

**UNIVERSIDADE NOVA DE LISBOA**

Faculdade de Ciências e Tecnologia

Departamento de Engenharia Electrotécnica

**Análise e Simulação de Técnicas de Análise de *Stocks***

**O Mercado de Retalho Como Caso de Estudo**

Por:

Bruno Miguel Marques Rodrigues

Dissertação apresentada na Faculdade de Ciências e Tecnologia da  
Universidade Nova de Lisboa para obtenção do grau de Mestre em  
Engenharia Electrotécnica e de Computadores

Orientador: Professor José Manuel Matos Ribeiro da Fonseca

**Lisboa**

**(2010)**

À memória do meu pai e à minha mãe,

À Alexandra e ao nosso futuro.

## Agradecimentos

---

Quero aqui agradecer às muitas pessoas que deram um importante contributo para eu chegar a esta fase da minha vida académica e pessoal, ao final deste trabalho e do curso.

Agradeço ao Professor José Manuel Matos Ribeiro da Fonseca pela oportunidade que me deu de realizar esta dissertação assim como pela orientação e conselhos, indispensáveis e muito importantes, prestados ao longo da mesma.

Agradeço à Eng. Inês Guerra, ao Mestre Fernando Moitinho e ao Mestre André Mora pela disponibilidade e apoio sempre demonstrados ao longo deste trabalho.

Aos meus colegas e amigos de dissertação, João Simões e Rogério Rebelo, pelo apoio, amizade e companheirismo demonstrado ao longo deste último ano.

Ao meu amigo de sempre, Ricardo Cardoso, por todo o apoio e amizade demonstrada ao longo dos últimos 12 anos, o que também contribuiu para eu chegar a esta fase. Agradeço, em especial, o apoio técnico e a amizade ao longo deste último ano.

Em particular e muito especialmente quero agradecer aos meus pais, João Rodrigues e Maria José Rodrigues, que me proporcionaram a frequência no ensino universitário e que sempre me apoiaram nas minhas decisões, nos bons e maus momentos pessoais e académicos. Ao meu pai, que embora já não esteja no meio de nós, eu sei que este era um sonho teu que estou prestes a concretizar. Estejas onde estiveres obrigado por tudo Pai.

Finalmente, agradeço à minha companheira dos últimos seis anos, Alexandra Cabrita, pelo apoio, força infindável que me deste ao longo destes anos e sem os quais teria sido mais difícil aqui chegar. Obrigado por acreditares sempre em mim, nas minhas capacidades, pela paciência quando eu não podia estar presente e por seres a companheira fantástica que és.

Nesta tese é efectuada a análise dos *stocks* de uma empresa de venda a retalho, sendo esta composta pela classificação dos produtos estudados quanto ao seu valor de uso e pela optimização dos stocks mínimos e máximos associados a cada produto.

O classificador dos produtos recorre ao método Análise ABC que permite o estudo do valor de uso de cada produto. Extraíndo-se um conjunto de dados de treino e de teste e através da utilização do método de classificação por árvores de decisão, obtêm-se as regras ou condições que formam o classificador em causa e que são utilizadas na aplicação desenvolvida.

A optimização dos *stocks* é feita através do modelo de optimização de *stocks*, designado por Modelo de Nível de Encomenda, que possibilita optimizar os *stocks* mínimos e máximos de cada produto o que por sua vez permite a minimização nos custos de *stocks* de muitos dos produtos. Assim é possível que a empresa em causa possa poupar dinheiro gasto na aquisição de produtos em quantidade excedentária.

Com a possibilidade de investimento em aplicações financeiras, é efectuado um estudo comparativo, para os produtos optimizados, que estima o dinheiro ganho, pela empresa utilizadora desta ferramenta, se usasse os *stocks* mínimos e máximos optimizados. É efectuada uma comparação gráfica para um produto escolhido entre a evolução dos *stocks* desse produto com os valores anteriormente praticados e os aconselhados pela ferramenta desenvolvida ao longo de um período escolhido.

Finalmente é apresentada uma pequena análise quanto à sazonalidade dos produtos.

# Abstract

---

This thesis analyzes the retail company's stocks, involving the construction of a classifier for the analyzed products based on their use value and involving the optimization of maximum and minimum stock for each one of the products analyzed.

The product's classifier uses a method called ABC Analysis that performs the product's study of its *use value*. Extracting a training set and a test set and with a classifying method called Decision Trees, classifier's rules are created and used in the developed tool.

The stock's optimization is performed by an optimization method called Order Level Method that computes the maximum and minimum stocks of each product. With this optimization it is possible to obtain stock's cost minimization for a large number of the analyzed products being possible, for the retail's company, to save the money spent in the acquisition of the analyzed products or in the surplus product's order quantities.

Thinking about financial investments, a comparative study is done showing the profit obtained by the company that uses this tool, using the new stock values. It is also possible, for the user, considering only one product for a user's specified time set, to perform a chart comparison between the stock's evolution using the old values and the new values optimized by the tool.

Finally a seasonality analysis of the products is presented.

# Índice de Matérias

Agradecimentos.....	2
Sumário.....	3
Abstract.....	4
Índice de Matérias.....	5
Índice de Figuras.....	7
Índice de Tabelas.....	8
Capítulo 1. Introdução.....	9
1.1. Motivação.....	9
1.2. Objectivos do Trabalho Desenvolvido.....	10
Capítulo 2. Conceitos.....	12
2.1. <i>Business Intelligence</i> .....	13
2.2. Gestão de Stocks – Modelos de Optimização.....	16
2.2.1. Modelos Determinísticos.....	19
2.2.1.1. Reposição Instantânea e Rotura Não Permitida.....	19
2.2.1.2. Reposição Instantânea e Rotura Permitida.....	22
2.2.1.3. Reposição Não Instantânea e Rotura Não Permitida.....	25
2.2.1.4. Reposição Não Instantânea e Rotura Permitida.....	27
2.2.2. Modelos Estocásticos.....	32
2.2.2.1. Revisão Cíclica.....	34
2.2.2.2. Nível de Encomenda.....	36
2.2.2.3. Política Mista.....	39
2.2.2.4. Revisão Cíclica e Nível de Encomenda – Vantagens e desvantagens.....	39
2.3. Análise ABC.....	39
Capítulo 3. Trabalho Desenvolvido.....	43
3.1. Introdução.....	43
3.2. Análise ABC – Classificador Implementado.....	44
3.2.1. Implementação.....	44
3.2.2. Apresentação de Resultados.....	46

3.3. Nível de Encomenda – Modelo de Análise Implementado.....	47
3.3.1. Implementação.....	48
3.3.2. Apresentação e Comparação de Resultados.....	50
3.3.2.1. Resultado da Implementação do Modelo Nível de Encomenda – Um Exemplo.....	50
3.3.2.2. Simulação de Ganhos Obtidos.....	51
3.4. Sazonalidade de Produtos.....	59
3.4.1. Implementação.....	59
3.4.2. Apresentação de Resultados.....	60
3.4.2.1. Sazonalidade por Estações do Ano.....	60
3.4.2.2. Sazonalidade por Mês.....	61
3.5. Software Desenvolvido ( <i>Stock Analysis Tool</i> ).....	62
Capítulo 4. Conclusões e Trabalho Futuro.....	66
Referências Bibliográficas.....	70
Anexo A. Breve Resumo de Especificações das Bases de Dados Utilizadas.....	72
Anexo B. Breve Descrição do <i>Web Service</i> Implementado.....	78
Anexo C. Distribuição Normal Reduzida.....	87
Anexo D. Modelos Estocásticos – Exemplos.....	88
Anexo E. Modelo Nível de Encomenda – Método Iterativo.....	91
Anexo F. Exemplo de Análise ABC.....	94
Anexo G. Regras da Árvore de Decisão Gerada.....	96

## Índice de Figuras

---

2.1 – Sistema de <i>Business Intelligence</i> : Estrutura ou Arquitectura usual.....	14
2.2 – Relação entre Quantidade Económica e diferentes Custos de Gestão de <i>Stocks</i> .....	18
2.3 – Reposição Instantânea e Rotura Não Permitida.....	20
2.4 – Reposição Instantânea e Rotura Permitida.....	22
2.5 – Reposição Não Instantânea e Rotura Não Permitida.....	25
2.6 – Rotura Não Instantânea e Rotura Permitida.....	28
2.7 – Revisão Cíclica: Tratamento de Encomendas.....	34
2.8 – Revisão Cíclica: Exemplo de Distribuição de Procura.....	35
2.9 – Nível de Encomenda: Tratamento de Encomendas.....	37
2.10 – Nível de Encomenda: Exemplo de Distribuição de Procura.....	37
2.11 – Curva ABC.....	40
3.1 – Árvore de Decisão Gerada.....	46
3.2 – Simulação de ganhos para o produto X.....	56
3.3 – Simulação de ganhos para todos os produtos.....	58
3.4 – Sazonalidade por Estação do Ano – Outono.....	60
3.5 – Sazonalidade por Mês.....	61
3.6 – <i>Stock Analysis Tool</i> – Form Inicial.....	63
3.7 – <i>Stock Analysis Tool</i> – Classificação dos produtos.....	64
3.8 - <i>Stock Analysis Tool</i> – Optimização de Todos os Produtos.....	64
3.9 - <i>Stock Analysis Tool</i> – Parametrizações.....	65
B.1 - Exemplo de Arquitectura Distribuída.....	78
C.1 – Exemplo de Distribuição Normal.....	87



## Índice de Tabelas

---

2.1 – Revisão Cíclica e Nível de Encomenda: Vantagens e Desvantagens.....	39
3.1 – Alguns dados trabalhados pertencentes ao conjunto de treino/teste.....	45
3.2 – Matriz de Confusão.....	46
C.1 – Tabela da Distribuição Normal Reduzida.....	87
F.1 – Exemplo de Análise ABC: Valor de Uso.....	94
F.2 – Exemplo de Análise ABC: Ordenação Decrescente por Valor de Uso.....	94
F.3 – Exemplo de Análise ABC: Classificação ABC dos Produtos.....	95

# CAPÍTULO 1

## Introdução

---

Este capítulo efectua a apresentação do trabalho realizado nesta tese e do contexto em que o mesmo se insere.

### 1.1. Motivação

No mercado da indústria retalhista, o *stock* de uma empresa corresponde normalmente a uma significativa fracção do capital financeiro investido. Os produtos em *stock* são existências que podem ter grande rotatividade ou não. No caso de produtos de baixa rotatividade, dependendo dos seus custos de aquisição, a empresa pode ter muito capital *parado* que poderia ser investido noutras áreas da empresa ou mesmo em activos financeiros que trariam certamente mais proveitos à mesma.

O controlo rigoroso dos *stocks* permitirá assim que o capital investido em produtos em relação aos quais se consigam ganhos com os novos níveis de *stock* otimizados possa ser aplicado de forma diferente e mais lucrativa. Esta é uma das razões para a utilização de uma ferramenta de apoio à decisão na optimização de *stocks*. Para além do aumento das margens de lucro, que muitas vezes é arriscado ou mesmo impossível para este tipo de empresas, tem-se a capacidade de *optimização* de espaço alocado em armazém que permite ter outros produtos em armazém ou aumentar a quantidade de produtos de elevada rotatividade que já se encontram alocados. Deste modo consegue-se então obter a minimização dos custos de *stock*.

Relativamente às indústrias retalhistas que se pensou como alvo do estudo nesta tese, podem considerar-se, entre outras, os hipermercados, as farmácias, armazéns de revenda e outras quaisquer empresas cujo negócio envolva venda a retalho.

É também importante haver a noção de quais os produtos mais importantes e os menos importantes relativamente aos retornos de vendas obtidos para estas indústrias. Os mais importantes correspondem a maiores retornos e os menos importantes correspondem a menor retorno. Os de menor retorno podem por vezes ter uma presença mínima em *stock*. É portanto importante construir uma funcionalidade que classifique os produtos relativamente ao valor financeiro que representam para a empresa.

No capítulo 2 mostram-se os conceitos em que se baseia este classificador assim como os diferentes modelos de *optimização de stocks* ([Costa, 2002], [Hillier, 2001], [Tavares, 1996]).

## 1.2. Objectivos do Trabalho Desenvolvido

As áreas de Tecnologias de Informação e os Sistemas de Suporte à Decisão são de grande interesse para o autor deste trabalho. Assim, e por forma a encontrar uma melhor solução/*optimização* para o problema em questão, decidiu-se implementar um sistema de apoio à decisão. Os objectivos principais deste trabalho são:

- Utilização de um classificador que possibilite ter-se uma maior sensibilidade da importância de cada produto;
- Estudo de diferentes modelos de *optimização de stocks*;
- Implementação de um dos modelos de *optimização de stocks* por forma a se minimizar os custos de *stocks* através da *optimização dos stocks* mínimo e máximo e quantidade a encomendar dos produtos considerados;

- estudo comparativo entre o modelo utilizado neste trabalho e o modelo utilizado no *software* actualmente utilizado num estabelecimento – ganhos/perdas que se obteriam utilizando a análise fornecida pelo *software* desenvolvido neste trabalho;
- análise de sazonalidade dos produtos para períodos de tempo pré-definidos ou definidos pelo utilizador.

O *software* de análise implementado, designado por SAT – *Stock Analysis Tool* – é fruto dos conhecimentos adquiridos na área da percepção sensorial assim como do profundo conhecimento e experiência, quer da área de modelos de suporte à decisão quer do mercado em que se insere este problema, do professor José Manuel da Fonseca. Como suporte dos objectivos definidos e respectiva demonstração de resultados foi desenhado um *software* na plataforma Visual Studio 2005 (linguagem de programação C#), assim como para a construção do classificador foi utilizado um *software* específico, nomeadamente, o CART.

# CAPÍTULO 2

## Conceitos

---

Ao longo do tempo a tecnologia sofreu grandes e rápidas transformações à medida que os mercados empresariais globais também evoluíram. A informação das empresas acerca dos seus negócios tornou-se indispensável assim como a troca de informação entre departamentos, clientes e/ou empresas. Deste modo, as Tecnologias de Informação (TI) são uma área indispensável para o sucesso das empresas no mercado em que se inserem.

Uma das áreas das TI mais importante, uma vez implementada, designa-se por *Business Intelligence* [Kimball, 2002]. Esta área, como se observa nas próximas secções, tem diversas áreas de aplicação, tais como, os mercados financeiros, as empresas de produção e os mercados de venda a retalho. Servindo de suporte aos gestores/directores de forma a que estes tomem as melhores decisões de negócio ou pelo menos as mais informadas. Embora já existam muitas soluções ao nível do controlo de *stocks* de empresas produtoras/fornecedoras, ao nível das empresas revendedoras a exploração deste mercado é diminuta, o que levou a que se desenvolvesse este trabalho.

Analisa-se então a contextualização desta tese ao nível do *Business Intelligence*, seguido dos Modelos de *optimização* de *stocks* estudados e do método de classificação de Produtos, Análise ABC.

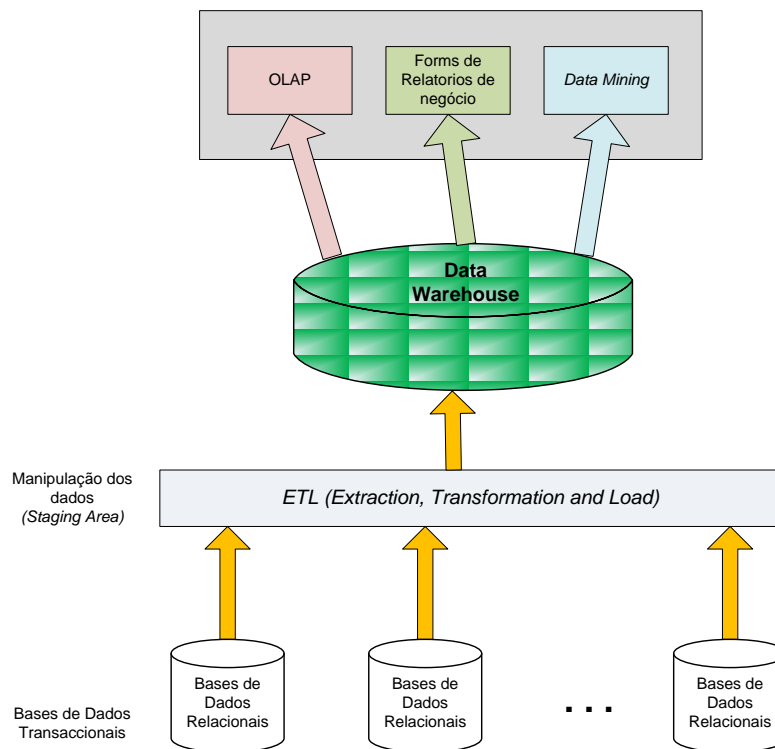
## 2.1. *Business Intelligence*

O *Business Intelligence* (BI) é uma das área das Tecnologias de Informação (TI ou IT, de *Information Technologies*) que engloba diversas tecnologias e aplicações que permitem o armazenamento, a integração, a análise e a apresentação de dados relativos a um negócio. Com o BI é possível [Gang, 2008]:

- processamento rápido de dados, ou seja, os dados podem ser acedidos a qualquer momento e processados rapidamente;
- análise inteligente e correlacionada da informação utilizando de forma total os modelos matemáticos aplicados fornecendo matéria científica e relevante ao nível da decisão;
- análise multidimensional, ou seja, diverso tipo de informação pode ser obtida e interligada tais como *merchandise*, fabricantes, fornecedores, preços, entre outros;
- resultados por análise progressiva. Através de uma análise contínua e de extracção progressiva de informação específica é possível conceber importantes relatórios de análise assim como indicadores de performance que servem como bom auxiliar de decisão.

De uma forma geral, os sistemas BI fornecem informação passada, presente e previsões para o negócio de acordo com a informação guardada até ao momento.

É assim importante ter conhecimento acerca da estrutura de um sistema BI. Apresenta-se na figura 2.1 uma possível estrutura para um sistema de *Business Intelligence* [Gang, 2008].



**Figura 2.1** – Sistema de *Business Intelligence*: Estrutura ou Arquitectura usual

Externamente ao sistema de BI, têm-se as bases de dados que estão distribuídas pelos diversos sectores da empresa e a partir dos quais, através do módulo ETL (*Extraction, Transformation and Load*), é extraída, tratada e carregada a informação necessária para obter os resultados pretendidos. Um *Data Warehouse* é um repositório onde é guardada a informação extraída das bases de dados externas e a partir do qual essa informação pode ser analisada, manipulada e tratada e assim atingir os resultados da análise BI pretendidos. Estes resultados podem ser obtidos através de diferentes aproximações:

- O *OnLine Analytical Processing* (OLAP) é uma aproximação que permite fornecer rápidas respostas a questões analíticas cuja origem consista em diferentes dimensões de análise; Esta característica multi-dimensional deu origem a um conceito designado por Cubo, que é constituído por *factos* também chamados por *medidas*, que por sua vez se encontram categorizados por *dimensões*;

- O *Data Mining* possibilita percorrer quantidades de dados muito elevadas e destes seleccionar a informação relevante. Recorrendo a esta informação e à análise lógica e estocástica da mesma é possível procurar e mostrar padrões (resultados) que auxiliem nas tomadas de decisão a fazer pelos responsáveis de negócio;
- Os *Forms* de Relatórios permitem apresentar os resultados de análise através de interfaces gráficas quer seja ao nível de aplicação de janelas ou aplicação *Web*.

O BI é assim muito utilizado ao nível das empresas financeiras como suporte aos seus gestores/directores para concepção de relatórios financeiros e de consolidação da empresa, *intelligence* direccionado aos clientes, gestão de risco, entre outras análises. Além da utilização em empresas financeiras, o BI tem também um papel muito importante na indústria de retalho.

### ***Business Intelligence* e a Indústria de Retalho**

Com o BI as indústrias de retalho conseguem ter mais e melhores noções de gestão relativamente ao planeamento de aquisição de material/mercadorias, do seu armazenamento, e preço, da publicidade a efectuar e como chegar aos clientes e direccionar os seus produtos às suas necessidades, ou seja, como obter melhores políticas de marketing. Grandes retalhistas (Wal-Mart, Foot Locker, Staples, Amazon.com entre outros) utilizam o BI e as suas ferramentas de análise por forma a definirem decisões estratégicas que passam por decidir o local para um *outlet*, qual a quantidade por cor ou tamanho de um produto que deve ser colocada à venda em determinada loja, quando e em quanto se deve implementar um desconto, entre outras decisões.

É também importante entender-se quais as razões que levaram a indústria de retalho a olhar para o BI com outros “olhos”. A sociedade actual é, a cada dia que passa, mais competitiva e o mercado do retalho não é diferente. A capacidade para aumentar as margens de lucro



tornou-se mais complexa, sendo esta uma das principais razões mesmo a principal razão para o recurso ao BI. É assim importante tirar o máximo de lucro dos custos de aquisição e dos espaços de armazenagem que muitas vezes ficam completamente lotados, ou seja, efectuar minimização de custos de *stock*. O crescimento acelerado do uso da internet em tão poucos anos foi também um bom incentivo pois abriu uma nova oportunidade de negócio podendo ser a internet vista como mais um local de vendas para os retalhistas.

O mercado do retalho, que inclui a indústria farmacêutica, encara assim o BI como uma área das tecnologias de informação muito importante de modo a conseguir resultados mais lucrativos para a empresa em questão.

Para se alcançar esta minimização do custo de *stocks* existem vários modelos de *optimização* de *stocks* que devem ser usados, isoladamente ou em conjunto, de acordo com as características do problema em estudo.

## 2.2. Gestão de Stocks – Modelos de Optimização

A gestão de *stocks* pode ser considerada como uma das áreas de aplicação do *Business Intelligence*. Nesta secção mostram-se alguns dos principais conceitos desse tipo de gestão.

Em primeiro lugar é importante perceber o conceito de *stock*. Por *stock* entende-se uma reserva de materiais ou produtos que tem como objectivo facilitar a produção ou venda de produtos conseguindo assim satisfazer a procura dos clientes [Costa, 2002]. De notar que muitas vezes ao *stock* de uma empresa dá-se o nome de *inventário*. Este nome, inventário, dado ao *stock* é dado de forma abusiva e por essa razão, ao longo deste documento será usada a palavra *stock*.

Para além de se conseguir proteger o negócio contra as incertezas da procura é ainda possível cobrir necessidades de trânsito de produtos, no que respeita à fabricação ou produção de

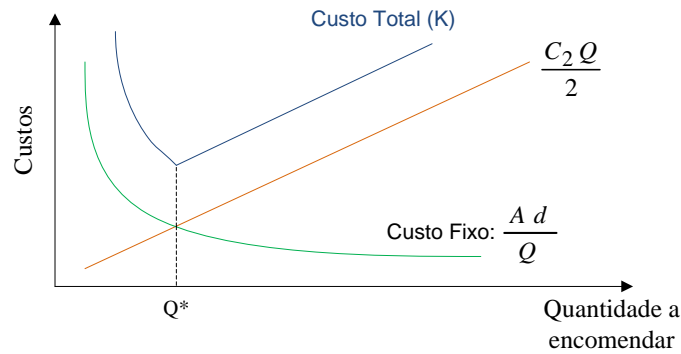
produtos, mais concretamente, trocas entre postos de trabalho, sendo ainda possível efectuar campanhas de descontos de quantidade.

A Gestão de *stocks* trabalha com diversos custos sendo um dos seus objectivos a minimização dos custos associados ao *stock* das empresas. Faz-se então aqui a apresentação de alguns dos custos a considerar assim como outras noções importantes.

A disponibilidade de um produto é a probabilidade de este estar disponível em *stock*, num determinado instante e quantidade, de forma a satisfazer os pedidos dos clientes. A esta grandeza, tal como se observará nas secções a seguir, é também dado o nome de nível de serviço, que convém ser o mais alto, mas também realista, possível.

O custo de encomenda é aquele associado à aquisição dos bens. Este tem grande influência nas quantidades óptimas a encomendar, como se verificará mais adiante. As roturas de *stock* são situações em que os pedidos dos clientes não são satisfeitos pois não existe quantidade suficiente de produto em *stock* para satisfazer esse pedido. Por isso se desejam evitar estas situações pois significam menos uma ou mais vendas não efectuadas, ou seja, perda de lucro. Deste modo existe um custo de rotura que é importante. O custo de manutenção de *stock* é igualmente importante pois os produtos têm que ser armazenados em locais e condições específicas estando deste modo dependentes da dimensão média de *stock*, influenciando a quantidade óptima a encomendar. De notar que todos os custos referidos são por unidade de tempo.

Existe uma relação entre os diferentes custos atrás referidos através da qual é possível determinar a quantidade *ótima* ou económica,  $Q^*$ , como se observa pela figura 2.2 [Costa, 2002].



**Figura 2.2** – Relação entre Quantidade Económica e diferentes Custos de Gestão de *Stocks*

A determinação da quantidade económica ou óptima de encomenda,  $Q^*$ , conduz à minimização do custo total do *stock* assim como faz com que o lucro marginal de um produto vendido seja igual à perda marginal relativa à não venda desse mesmo produto. É assim importante saber o que se entende por lucro e perda, noções estas que se podem observar nas equações 2.1 e 2.2.

$$\text{Lucro} = \frac{\text{Preço Unitário de venda}}{\text{de venda}} - \frac{\text{Custo Unitário de Aquisição}}{\text{de Aquisição}} \quad (2.1)$$

$$\text{Perda} = \text{Custo Unitário} - \text{Valor residual Unitário} \quad (2.2)$$

Como se verificará nas secções seguintes é possível fazer gestão de *stocks* para situações determinísticas, em que a procura é fixa e conhecida, assim como para situações estocásticas, em que a procura não é conhecida sendo portanto variável. Para cada uma destas situações existem vários modelos para se conseguir a *optimização* de *stocks* que se pretende alcançar com a gestão de *stocks*.

A optimização de *stocks* baseia-se na aplicação de diferentes políticas de reaprovisionamento (modelos estocásticos) ou de análises determinísticas de *stocks* (modelos determinísticos). É assim necessário ter conhecimento da natureza do problema no que respeita ao tipo de procura, se esta é estocástica ou determinística.

Deste modo, os modelos dividem-se em dois tipos: Modelos Determinísticos e Modelos Estocásticos.

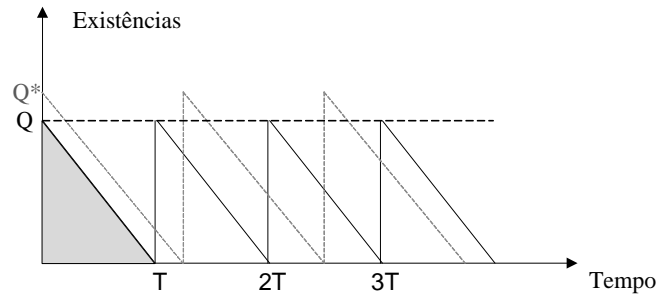
### 2.2.1. Modelos Determinísticos

Nestes modelos a procura e o tipo de fornecimento considerados são constantes e previamente conhecidos, existindo quatro modelos determinísticos possíveis que diferem relativamente a tipos de reposição e admissibilidade de rotura de *stocks* [Hillier, 2001], [Costa, 2002]:

- Reposição Instantânea e Rotura Não Permitida com procura e fornecimento determinísticos;
- Reposição Instantânea e Rotura Permitida com procura e fornecimento determinísticos;
- Reposição Não Instantânea e Rotura Não Permitida com procura determinística e constante e fornecimento constante nos períodos de fornecimento;
- Reposição Não Instantânea e Rotura Permitida com procura determinística e fornecimento constante nos períodos de fornecimento.

#### 2.2.1.1. Reposição Instantânea e Rotura Não Permitida

Este modelo é implementado quando a reposição dos produtos é instantânea e a quantidade de produto em *stock* não admite roturas, ou seja, a rotura de *stocks* no negócio em causa tem um impacto demasiado relevante. Como se pode observar pela figura 2.3, na linha negra tem-se a situação ideal em que com a quantidade a encomendar,  $Q$ , ao fim de um tempo  $T$  o *stock* seria nulo e então era feita e recebida instantaneamente uma nova encomenda com  $Q$  unidades. Ou seja, a quantidade encomendada é consumida a uma taxa constante ao longo de um tempo  $T$ . Da figura 2.3 conclui-se ainda que quanto maior for a quantidade ótima a encomendar, também designada por quantidade ótima ( $Q^*$ ), maior será o intervalo entre encomendas.



**Figura 2.3** – Reposição Instantânea e Rotura Não Permitida

Tal como se pode ver pela figura 2.3 é necessário ter conhecimento do valor da quantidade encomendada ( $Q$ ) que depende do intervalo de tempo entre encomendas ( $T$ ) e da procura por unidade de tempo ou taxa de procura ( $d$ ) como se pode observar pela equação 2.3.

$$Q = T \times d \quad (2.3)$$

A encomenda é a acção que rege todo este processo sendo deste modo necessário começar por saber o valor associado ao custo de encomenda (equação 2.4), custo este que é calculado com base no custo associado ao processamento da encomenda ( $A$ ), que se admite independente da quantidade encomendada, no custo unitário do produto ( $C_1$ ) e na quantidade encomendada. De notar que a parcela ( $C_1Q$ ) designa-se por custo de aquisição.

$$\text{custo de encomenda} = A + C_1Q \quad (2.4)$$

A manutenção dos *stocks* é vital de forma a que os produtos estejam sempre bem acondicionados, ou seja, que não se estraguem, existindo assim um custo de posse. O custo de posse diz respeito a capital imobilizado devido a aquisição de encomendas, ao aluguer de armazém, vencimentos com pessoal de armazém, electricidade, entre outras despesas. Como se pode observar pela equação 2.5, depende do custo de manter em *stock* ( $C_2$ ) uma unidade de produto e do nível médio de existências ao longo de um ciclo de tempo  $T$ .

$$\text{custo de posse} = C_2 \times \frac{Q}{2} \times T \quad (2.5)$$

De notar ainda que  $\frac{Q \times T}{2}$  corresponde à área sombreada presente na figura 2.3.

Deste modo, tendo o valor do custo de encomenda e do custo de posse já é possível saber o valor do custo total de armazenamento por ciclo,  $C_T$ , presente na equação 2.6.

$C_T$  = custo de encomenda + custo de posse

$$C_T = (A + C_1 Q) + \left( C_2 \times \frac{Q}{2} \times T \right) \quad (2.6)$$

Os custos até aqui calculados são custos totais mas a comparação de custos deve ser feita por unidade de tempo. Deste modo define-se um novo custo designado por custo por unidade de tempo ( $K$ ) que se encontra definido pela equação 2.7.

$$K = \frac{C_T}{T} \Leftrightarrow \dots \Leftrightarrow K = \frac{A d}{Q} + C_1 d + C_2 \frac{Q}{2} \quad (2.7)$$

Assim, o custo  $K$  encontra-se expresso em função de uma única variável,  $Q$ . De forma a se determinar o valor de  $Q$  que minimiza  $K$ , também designada por quantidade económica de encomenda ( $Q^*$ ), matematicamente basta derivar  $K$  em ordem a  $Q$  e, seguidamente, igualar a zero, tal como se observa pela equação 2.8. A expressão que determina o valor desejado de  $Q$  designa-se por Fórmula do Lote Económico ou Fórmula de *Wilson* [Costa, 2002].

$$\begin{aligned} \frac{\partial K}{\partial Q} = 0 &\Leftrightarrow -\frac{A d}{Q^2} + \frac{C_2}{2} = 0 \Leftrightarrow \\ &\Leftrightarrow Q^* = \sqrt{\frac{2 A d}{C_2}} \end{aligned} \quad (2.8)$$

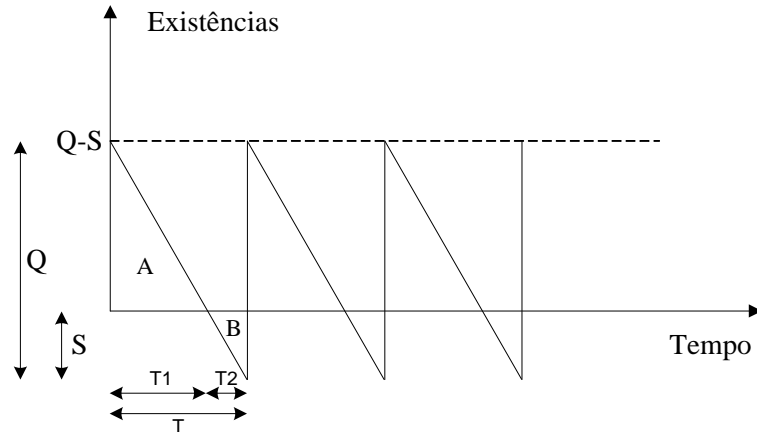
Sendo  $A$ ,  $d$  e  $Q$  constantes positivas então é possível concluir-se que a segunda derivada de  $K$  em ordem a  $Q$  é sempre positiva, ou seja, é uma função cuja concavidade da sua curva está sempre *voltada para cima* admitindo deste modo um único mínimo. Este valor mínimo único corresponde então ao custo total mínimo por unidade de tempo ( $K^*$ ), referente à quantidade económica de encomenda.

$$K^* = C_1 d + \sqrt{2 C_2 A d} \quad (2.9)$$

Destes resultados podemos retirar algumas conclusões que permitem uma melhor compreensão das decisões que possam ser tomadas. O valor de  $Q^*$  é proporcional ao valor da procura  $d$ , tal como esperado enquanto que, de forma contrária, quanto maior for o custo de armazenagem,  $C_2$ , menor é o valor de  $Q^*$ , ou seja, quanto menor for esse custo maior é a quantidade a encomendar e vice-versa. De notar ainda que para o cálculo de  $Q^*$  não é contabilizado o custo unitário de aquisição,  $C_1$ , e deste modo ao assumir-se que não é possível existir roturas existe o compromisso de se ter sempre disponível o artigo em causa, ou seja, o custo unitário de aquisição, para este modelo, é irrelevante.

### 2.2.1.2. Reposição Instantânea e Rotura Permitida

Neste caso, como se pode observar pela figura 2.4, a reposição dos produtos é considerada instantânea mas são admitidas roturas nos *stocks* dos produtos transaccionados.



**Figura 2.4** – Reposição Instantânea e Rotura Permitida

Por  $Q$  continua-se a designar a quantidade encomendada, tendo uma nova grandeza,  $S$ , que diz respeito ao *stock* em falta ou procura não satisfeita. Durante o período  $T_1$ , em que a procura é satisfeita, os produtos são consumidos ou vendidos a uma taxa constante até que o seu *stock* entra em rotura (número de existências menor que zero). A partir deste momento e ao longo de um período  $T_2$ , designado por período de penúria, não é possível satisfazer a

procura até que é recepcionada uma nova encomenda de quantidade  $Q$  que eleva os níveis de *stock* do produto até a um valor  $Q-S$ .

A quantidade a encomendar,  $Q$ , e o custo de processamento de encomenda são calculados através das equações 2.3 e 2.4, mostradas no modelo anterior. De notar apenas que, no caso deste modelo,  $Q$  é fornecida de forma instantânea a intervalos de tempo fixos  $T=T_1+T_2$ , também designados por ciclos de reaprovisionamento.

O custo de posse depende do custo de manter uma unidade em *stock* ( $C_2$ ) e do nível médio de existências  $\frac{Q-S}{2}$  ao longo de  $T_1$ , como se pode observar pela equação 2.10. A parcela  $\frac{Q-S}{2} \times T_1$  corresponde à área  $A$  que se pode observar na figura 2.4.

$$\text{custo de posse} = C_2 \times \frac{Q-S}{2} \times T_1 \quad (2.10)$$

Admitindo-se roturas de *stock* dos produtos tem-se um novo custo a considerar designado por custo de rotura, que se mostra na equação 2.11. Este custo depende do custo por unidade em falta,  $C_3$ , do nível médio de unidades em falta,  $\frac{S}{2}$ , ao longo do período de penúria,  $T_2$ .

$$\text{custo de rotura} = C_3 \times \frac{S}{2} \times T_2 \quad (2.11)$$

Na posse dos valores dos custos de encomenda, de posse e de rotura é possível determinar o custo total de armazenamento por ciclo ( $C_T$ )

$$C_T = \text{custo de encomenda} + \text{custo de posse} + \text{custo de rotura}$$

Durante o período de tempo  $T_1$  são consumidas  $(Q-S)$  unidades, como já se tinha visto anteriormente. Deste modo,  $Q-S = d \times T_1$  e de forma análoga tem-se que  $S = d \times T_2$  e assim, fazendo as substituições das variáveis  $T_1$  e  $T_2$  nas equações 2.10 e 2.11, obtém-se a equação 2.12 que define  $C_T$ .

$$C_T = (A + C_1 Q) + C_2 \times \frac{(Q-S)^2}{2d} + C_3 \times \frac{S^2}{2d} \quad (2.12)$$



Pela equação 2.3 e substituindo T pode-se determinar o custo por unidade de tempo, K, definido em função de Q e S (equação 2.13).

$$K = \frac{C_T}{T}$$

$$K = \frac{A d}{Q} + C_1 d + C_2 \times \frac{(Q - S)^2}{2 Q} + C_3 \times \frac{S^2}{2 Q} \quad (2.13)$$

K encontra-se assim expresso em função de Q e S. De forma a determinar os valores de Q e S que minimizam K tem que se derivar K em ordem a Q e S e, em seguida, igualar ambas as derivadas a zero obtendo-se assim um sistema de duas equações que resultam nos valores económico ou óptimo de Q, Q\*, e de S, S\*.

$$\begin{cases} \frac{\partial K}{\partial Q} = 0 \\ \frac{\partial K}{\partial S} = 0 \end{cases}, \text{sendo } C_3 \neq 0 \quad \Leftrightarrow \dots \Leftrightarrow$$

$$Q^* = \sqrt{\frac{2 A d}{C_2}} \times \sqrt{\frac{C_2 + C_3}{C_3}} \quad (2.14)$$

$$S^* = \sqrt{2 A d C_2} \times \frac{1}{\sqrt{C_3(C_2 + C_3)}} \quad (2.15)$$

Substituindo Q\* e S\* em K, obtém-se o custo total mínimo por unidade de tempo, K\*:

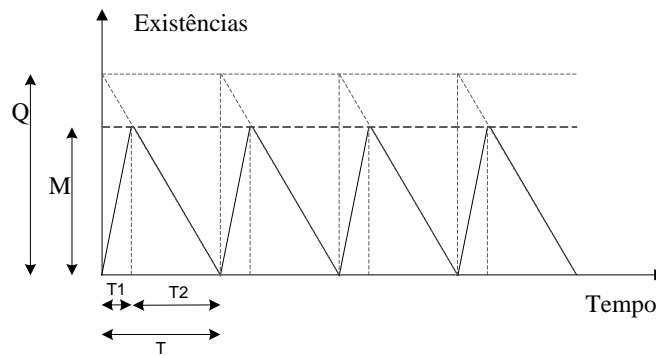
$$K^* = C_1 d + \sqrt{2 A d C_2} \times \sqrt{\frac{C_3}{C_2 + C_3}} \quad (2.16)$$

Pode-se então tirar algumas conclusões relativas a este modelo e até compará-lo com o modelo anterior. O custo por unidade em falta, C<sub>3</sub>, mostra ter muita importância no que diz respeito a este modelo. Aumentos de C<sub>3</sub> levam à diminuição em simultâneo de Q\* e S\*, que se vão aproximando de Q\* do modelo anterior e de zero, respectivamente. No caso de se estar perante valores muito baixos de C<sub>3</sub>, ou seja, quando as situações de rotura são pouco

penalisadas, os valores de  $Q^*$  e  $S^*$  aumentam significativamente, sendo assim possível efectuar maiores encomendas com maiores intervalos de tempo entre encomendas.

### 2.2.1.3. Reposição Não Instantânea e Rotura Não Permitida

Enquanto que nos casos apresentados anteriormente, a reposição do *stock* era instantânea, neste caso considera-se que existe um período de reposição de *stock*,  $T_1$ , tendo assim uma taxa de reposição de *stock*,  $p$ , atingindo no final da reposição do *stock* um nível máximo de *stock*,  $M$ , como se pode observar na figura 2.5.



**Figura 2.5** – Rotura Não Instantânea e Rotura Não Permitida

De notar que o custo de encomenda é dado pela equação 2.4.  $Q$  é também fornecida a intervalos de tempo fixos,  $T = T_1 + T_2$ . Ao longo do período  $T_1$  ocorre tanto procura como reposição, logo o *stock* aumenta à taxa  $(p-d)$ . Durante o período,  $T_2$ , em que só existe satisfação da procura, o *stock* diminui à taxa de procura  $d$ .

O *stock* máximo,  $M$ , é dado pela equação 2.17 e através desta calcula-se a quantidade encomendada,  $Q$  (equação 2.18).

$$M = T_1 \times (p - d) = T_2 \times d \quad (2.17)$$

$$Q = d \times T = d \times (T_1 + T_2) = p \times T_1 \quad (2.18)$$

Da equação 2.18 extrai-se o valor do período de reposição de stock  $T_1$ .

$$T_1 = \frac{Q}{p} \quad (2.19)$$

Das equações 2.17 e 2.19 encontra-se uma nova expressão para o *stock* máximo:

$$M = Q \left( 1 - \frac{d}{p} \right) \quad (2.20)$$

Das equações anteriores extrai-se o período em que existe apenas consumo, ou seja, a  $T_2$ :

$$T_2 = \frac{M}{d} \quad (2.21)$$

Observe-se que  $M$  é sempre menor do que  $Q$  e quanto maior for  $p$  relativamente a  $d$ , mais próximos serão os valores de  $Q$  e  $M$ .

É possível agora fazer a análise do custo total de armazenamento por ciclo,  $C_T$ . Não estando perante a admissibilidade de roturas no *stock* dos produtos, este custo apenas depende do custo de encomenda e do custo de posse.

$$C_T = \text{custo de encomenda} + \text{custo de posse}$$

É assim necessário ter conhecimento acerca do modo como extrair o valor do custo de posse.

A equação 2.22 mostra como se pode fazer a extracção deste valor, sendo  $\frac{M}{2}$  o nível médio de existências e  $C_2$  o custo de manter uma unidade de artigo em stock, ao longo de um ciclo de tempo  $T$ .

$$\text{Custo de posse} = C_2 \times \frac{M}{2} \times T \quad (2.22)$$

É assim possível apresentar o custo total de armazenamento por ciclo, tal como se pode observar pela equação 2.23.

$$C_T = A + C_1 Q + C_2 \left( 1 - \frac{d}{p} \right) \frac{T}{2} \quad (2.23)$$

Pode-se também determinar desde já a expressão relativa a custo por unidade de tempo,  $K$ .

$$K = \frac{C_T}{T} \Leftrightarrow \dots \Leftrightarrow K = \frac{A d}{Q} + C_1 d + C_2 \frac{Q}{2} \left(1 - \frac{d}{p}\right) \quad (2.24)$$

Tal como no primeiro modelo analisado, o custo  $K$  é apenas dependente da variável  $Q$ . Deste modo, de forma a calcular a quantidade económica,  $Q^*$ , que minimiza o custo por unidade de tempo deriva-se o custo por unidade de tempo em ordem a  $Q$  e, seguidamente, iguala-se a zero, sabendo que o ciclo corresponde a  $\frac{Q}{d}$ .

$$\frac{\partial K}{\partial Q} = 0 \Rightarrow \dots \Rightarrow Q^* = \sqrt{\frac{2 A d}{C_2}} \times \sqrt{\frac{p}{p - d}} \quad (2.25)$$

Estando na posse do valor da quantidade económica (equação 2.25) é então possível obter a expressão do custo total mínimo por unidade de tempo,  $K^*$ .

$$K^* = C_1 d + \sqrt{2 A d C_2} \times \sqrt{\frac{p - d}{p}} \quad (2.26)$$

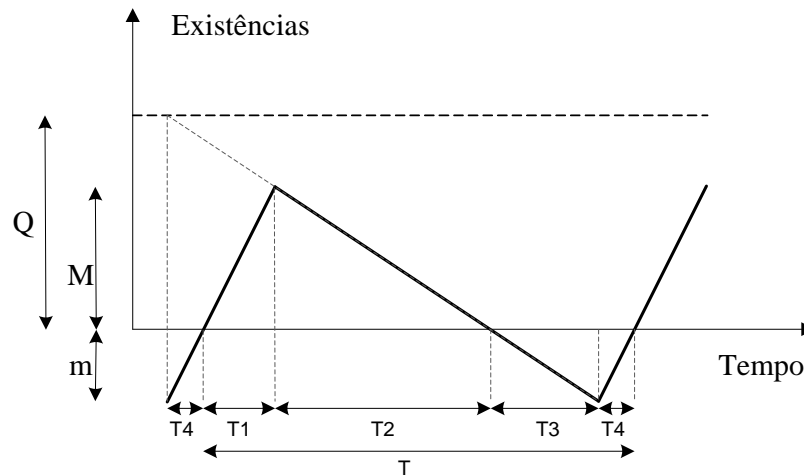
Comparando as equações deste modelo e do primeiro analisado, ou seja, as equações 2.26 e 2.9, respectivamente, conclui-se que o  $K^*$  deste modelo é sempre inferior ao do primeiro modelo analisado.

Constata-se ainda que à medida que se aumenta a taxa de reposição do *stock*,  $p$ , ou de outra forma, quanto mais a reposição se aproxima de ser instantânea, o valor de  $Q^*$  deste modelo reduz de igual modo aproximando-se do valor de  $Q^*$  verificado no primeiro modelo analisado. Relativamente ao custo total mínimo deste modelo, este aumenta aproximando-se do custo total mínimo verificado no primeiro modelo.

#### 2.2.1.4. Reposição Não Instantânea e Rotura Permitida

Tal como em todos os casos analisados até agora também este modelo está sujeito a uma procura constante e determinística e tal como no caso anterior a reposição não é instantânea. Deste modo, está sujeito a uma taxa de reposição,  $p$ , e a uma taxa de procura,  $d$ . Este facto

pode ser observado na figura 2.6 assim como o facto de, no caso deste modelo, se ter quatro períodos de análise diferentes. Deste modo é importante fazer algumas observações relativas a esses mesmos períodos e respectivas características da *curva* de evolução dos *stocks*.



**Figura 2.6** – Rotura Não Instantânea e Rotura Permitida

No período  $T_1$ , o *stock* varia de forma crescente e constante de acordo com uma taxa de variação igual a  $(p-d)$  sendo  $p$  maior que  $d$ . Verificam-se reposição de *stock* e consumos satisfeitos.

No período  $T_2$ , o *stock* varia de forma decrescente e constante de acordo com uma taxa de variação igual a  $d$ . Não existe qualquer reposição de *stock* e o consumo é sempre satisfeito.

No período  $T_3$ , o *stock* varia da mesma forma que no período anterior. Não se verificam reposições de *stock*. De notar que o *stock* negativo corresponde a procura não satisfeita nesse período sendo estas unidades encomendadas consumidas no período  $T_4$ .

No período  $T_4$ , o *stock* varia de forma crescente e constante de acordo com uma taxa de variação igual  $(p-d)$  em que é satisfeita a procura verificada no período  $T_3$ .

De notar que também para este modelo o custo de encomenda é determinado com recurso à equação 2.4 e que relativamente ao ciclo de tempo de análise,  $T$ , este é determinado por  $T = T_1 + T_2 + T_3 + T_4$ .

Analisando a figura 2.6 e tendo em conta as considerações anteriores é possível tirar imediatamente algumas conclusões. Os períodos  $T_1$  e  $T_2$  encontram-se expressos nas equações 2.27 e 2.28, respectivamente. Com base nestes dois resultados, encontram-se as expressões do nível máximo de *stock* (equações 2.29 e 2.30),  $M$ , e do nível mínimo de *stock* (equações 2.31 e 2.32),  $m$ .

$$T_1 = \frac{Q}{p} \quad (2.27) \qquad T_2 = \frac{M}{d} \quad (2.28)$$

$$M = T_1(p - d) \quad (2.29) \qquad m = T_4(p - d) \quad (2.31)$$

$$M = T_2 d \quad (2.30) \qquad m = T_3 d \quad (2.32)$$

$$M + m = Q - d(T_1 + T_4) \quad (2.33)$$

Com base nas equações apresentadas em cima obtém-se a relação entre os quatro períodos de análise e a procura,  $p$ , e a reposição,  $d$ , como se pode observar na equação 2.34.

$$\frac{T_1}{T_2} = \frac{T_4}{T_3} = \frac{d}{p - d} \quad (2.34)$$

É assim possível extrair-se uma nova expressão relativa à quantidade encomendada,  $Q$ .

$$Q = T \times d = (T_1 + T_2 + T_3 + T_4) d = \frac{p \times d}{p - d} (T_2 + T_3) \quad (2.35)$$

É assim possível, neste momento, determinar a expressão do custo total de armazenamento por ciclo,  $C_T$ . No caso deste modelo, é definido pelos custos de encomenda, de posse e de rotura, tal como no segundo modelo. Deste modo, é necessário encontrar as expressões do custo de posse e do custo de rotura. O custo de posse apenas se relaciona com os períodos de tempo,  $T_1$  e  $T_2$ , pois na situação de posse há sempre consumo satisfeito que corresponde a estes dois períodos, o que leva à equação 2.36. O custo de rotura, tal como o próprio nome indica, apenas se relaciona com os períodos em que o consumo não é satisfeito, ou seja, os períodos  $T_3$  e  $T_4$ , o que leva à equação 2.37.

$$\text{custo de posse} = C_2 \times \frac{M}{2} \times (T_1 + T_2) \quad (2.36)$$

$$\text{custo de rotura} = C_3 \times \frac{m}{2} \times (T_3 + T_4) \quad (2.37)$$

É então possível mostrar como calcular o custo total de armazenamento por ciclo,  $C_T$ , tal como mostra a equação 2.38.

$C_T$  = custo de encomenda + custo de posse + custo de rotura

$$C_T = A + C_1 Q + C_2 \times \frac{M}{2} \times (T_1 + T_2) + C_3 \times \frac{m}{2} \times (T_3 + T_4) \quad (2.38)$$

Tal como nos modelos anteriores, sabendo que o custo por unidade de tempo,  $K$ , é dado por  $K = \frac{C_T}{T}$  e que  $T = \frac{Q}{d}$  então é possível determinar o seu valor (equação 2.39).

$$K = \frac{A + C_1 Q + C_2 \times \frac{M}{2} \times (T_1 + T_2) + C_3 \times \frac{m}{2} \times (T_3 + T_4)}{\frac{Q}{d}} \quad (2.39)$$

Conclui-se então que o custo por unidade de tempo,  $K$ , depende das grandezas  $Q$ ,  $m$ ,  $T_1$ ,  $T_2$ ,  $T_3$  e  $T_4$ . Seguindo o procedimento usado nos modelos anteriores, é possível definir as expressões de  $Q$ ,  $m$ ,  $T_1$ ,  $T_2$ ,  $T_3$  e  $T_4$  que minimizam o custo por unidade de tempo, expressões estas que correspondem às equações mostradas a seguir.

$$Q^* = \sqrt{\frac{2 A d}{C_2}} \times \sqrt{\frac{p}{p-d}} \times \sqrt{\frac{C_2 + C_3}{C_3}} \quad (2.40)$$

$$m^* = \sqrt{2 A d C_2} \times \sqrt{\frac{p-d}{p}} \times \sqrt{\frac{1}{(C_2 + C_3)C_3}} \quad (2.41)$$

Sendo  $T^* = \frac{Q^*}{d}$  obtém-se assim:

$$T_2^* = \sqrt{\frac{2 A}{C_2 d}} \times \sqrt{\frac{p-d}{p}} \times \sqrt{\frac{C_3}{C_2 + C_3}} \quad (2.42)$$

$$T_3^* = \sqrt{\frac{2 A}{C_3 d}} \times \sqrt{\frac{p-d}{p}} \times \sqrt{\frac{C_2}{C_2 + C_3}} \quad (2.43)$$

Destes dois últimos resultados são retirados os valores de  $T_1^*$  e  $T_4^*$ , respectivamente.

$$T_1^* = T_2^* \times \frac{d}{p-d} \quad (2.44)$$

$$T_4^* = T_3^* \times \frac{d}{p-d} \quad (2.45)$$

Com recurso às equações anteriores é possível determinar o custo total mínimo por unidade de tempo,  $K^*$ .

$$K^* = C_1 d + \sqrt{2 A d C_2} \times \sqrt{\frac{p-d}{p}} \times \sqrt{\frac{C_3}{C_2 + C_3}} \quad (2.46)$$

A primeira conclusão, comparando com o primeiro modelo analisado, refere-se ao pormenor de  $Q^*$  determinado neste modelo ser igual à expressão obtida para  $Q^*$ , no primeiro modelo, multiplicando a segunda parcela deste último por dois factores *correctivos*. O primeiro factor,  $\sqrt{\frac{p-d}{p}}$ , sempre maior que 1, tende para 1 à medida que a reposição se torna instantânea ( $p$  aumenta) aproximando-se à expressão determinada no primeiro modelo. De seguida, se  $C_3$  aumentar, ou seja, quando a ocorrência de roturas se torna inadmissível para o sistema em causa, o segundo factor,  $\sqrt{\frac{C_3}{C_2 + C_3}}$ , que tal como o primeiro factor é sempre maior que 1, tenderá para 1 o que resulta na expressão obtida para  $Q^*$  no primeiro modelo analisado. Assim, com estes dois factores é possível concluir que a quantidade económica praticada neste modelo é sempre superior à quantidade económica obtida no primeiro modelo analisado.

Comparando com o segundo modelo analisado conclui-se que os resultados são coincidentes no caso de se aumentar muito  $p$ , ou seja, caso a reposição se aproxime do instantâneo.

Aumentando apenas  $C_3$  implica uma coincidência de resultados com o terceiro modelo analisado anteriormente.



Relativamente à dimensão da encomenda e ao custo total por unidade de tempo é possível concluir que à medida que  $C_3$  e  $p$  aumentam verifica-se a diminuição da quantidade encomendada e o aumento no custo total por unidade de tempo.

### 2.2.2. Modelos Estocásticos

Nos modelos estocásticos, ao contrário dos analisados anteriormente, tem-se como característica principal a procura ser desconhecida, variando segundo uma distribuição normal. Existem nestes modelos três políticas de reaprovisionamento: Revisão Cíclica; Nível de Encomenda e Política Mista.

As políticas de reaprovisionamento disponibilizam informação ao nível das decisões a tomar, como por exemplo, quantidade a encomendar e em que momento encomendar e ao nível do objectivo final de uma *optimização de stocks*, ou seja, escolher a solução que corresponda a um custo mínimo.

Se a procura dos produtos é variável então é importante, ao nível da gestão, que seja mantido um adequado nível de serviço. É assim importante perceber, antes de mais, em que consiste este novo conceito. No caso da gestão de *stocks*, nível de serviço é o complemento da probabilidade de se ter rotura no *stock*. Como exemplo, admitindo 5% de falhas tem-se um nível de serviço de 95%. Uma forma de determinar o valor do nível de serviço de um determinado produto é mostrado na equação 2.47.

$$\text{Nível de Serviço} = 1 - \frac{\text{Procura Anual Não Satisfeita}}{\text{Procura Anual}} \quad (2.47)$$

A incerteza quanto à procura dos produtos faz com que haja sempre possibilidades de ocorrência de roturas de *stock*, sendo deste modo necessário reduzir a ocorrência de roturas, redução esta que é possível mantendo unidades extra de produto em *stock*. A esta quantidade extra de produto dá-se o nome de *stock de segurança*. Existe inclusivamente uma relação entre

estes dois parâmetros e outros parâmetros associados às diferentes políticas de reaprovisionamento aqui analisadas, isto é, um aumento do nível de serviço corresponde a um aumento do *stock* de segurança que, por sua vez, corresponde a um aumento do nível de encomenda ou a um aumento do nível máximo, consoante a política de reaprovisionamento adoptada.

Um conceito determinante para o sucesso das políticas de reaprovisionamento é o tempo de reposição dos produtos por parte dos fornecedores. O tempo de reposição dos produtos consiste no intervalo de tempo entre o pedido da encomenda ao fornecedor e a recepção dos produtos encomendados por parte do sistema do cliente.

No entanto, outras condições têm que ser garantidas de forma a que os modelos possam ser aplicados. Os *stocks* são continuamente *vigiados*, ou de outra forma, sempre que haja uma entrada ou saída de produtos do armazém as existências devem ser recalculadas sendo deste modo sempre possível saber a quantidade tida em *stock* para cada produto. A cada encomenda está associado um custo de processamento de encomenda,  $A$ , que deve ser fixo e que pode envolver, entre outros factores, percentagem do ordenado associado ao tempo necessário para processar uma encomenda, percentagem do valor dos gastos relativos ao computador em que se efectua a encomenda, etc. Tal como nos outros modelos analisados anteriormente o custo da encomenda é proporcional à quantidade encomendada (ver equação 2.4).

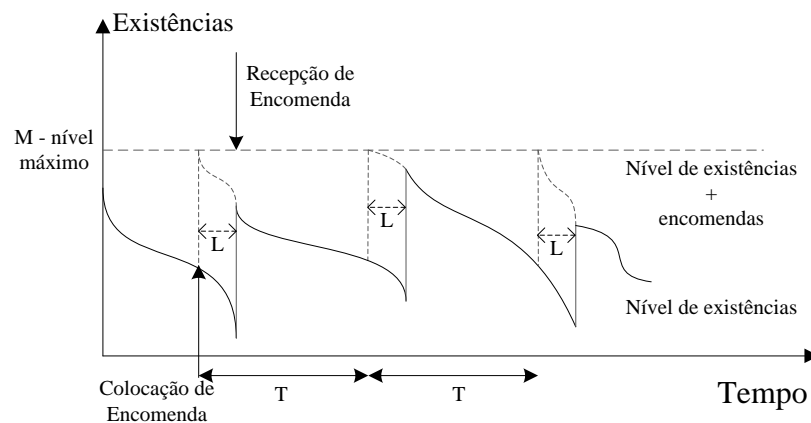
O desvio padrão da média da procura é calculado segundo a equação 2.48, sendo  $X_i$  uma observação de procura,  $\bar{X}$  a média da procura e  $N$  o número total de amostras.

$$\text{desvio padrão} = \sqrt{\frac{(X_i - \bar{X})^2}{N}} \quad (2.48)$$

### 2.2.2.1. Revisão Cíclica

Neste modelo, as encomendas são colocadas a intervalos fixos de tempo (intervalo de encomenda) sendo a quantidade a encomendar variável de forma a colocar o *stock* num nível máximo,  $M$ . Pela figura 2.7 é possível observar-se como é efectuado o tratamento das encomendas e o tipo de evolução que se tem nos *stocks* considerando esta política de reaprovisionamento.

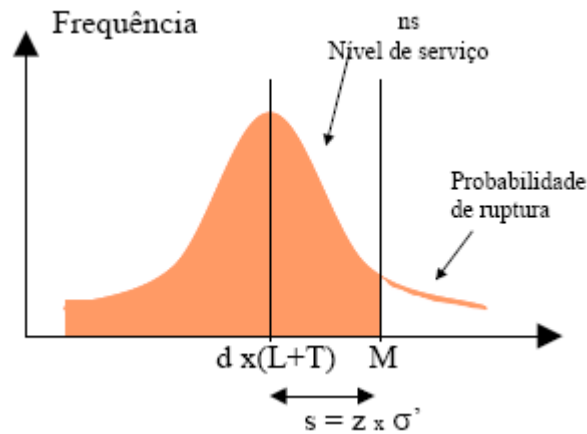
Entre os vários pontos de revisão há sempre a possibilidade de ocorrer rotura de *stock* devido ao tempo de reposição dos produtos ser desconhecido, pois depende da capacidade do fornecedor colocar as encomendas em tempo útil. Como consequência, pode não ser possível satisfazer o pedido dos clientes que procurem esse produto.



**Figura 2.7** – Revisão Cíclica: Tratamento de Encomendas

Para iniciar a análise de *optimização* é necessário ter conhecimento acerca da procura por unidade de tempo,  $d$ , o número esperado de unidades de tempo para entrega,  $L$ , o número de unidades de tempo entre revisões do *stock*,  $T$ , e o *stock* de segurança,  $s$ .

Comece-se então pelo *stock* de segurança a considerar. Embora já se tenha feito uma pequena introdução relativamente a este parâmetro, o seu cálculo varia com a política de reaprovisionamento utilizada. No caso da revisão cíclica, a figura 2.8 ilustra como analisar o *stock* de segurança já que este é determinado através da distribuição da procura do produto.



**Figura 2.8:** Revisão Cíclica: Exemplo de distribuição de procura

O *stock* de segurança é calculado através do desvio padrão da procura,  $\sigma'$ , e do acontecimento relativo ao nível de serviço (ns) respectivo,  $z$ , segundo as seguinte expressões.

$$\sigma' = \sqrt{L + T} \times \sigma_{u.t.} \quad (2.49)$$

$$s = z \times \sigma' \quad (2.50)$$

De notar que  $z$  é determinado recorrendo à tabela da distribuição normal tida no Anexo C deste documento e que  $\sigma_{u.t.}$  se refere ao desvio padrão relativo à unidade de tempo de reaprovisionamento.

É então possível mostrar as relações necessárias à *optimização* dos *stocks* utilizando este modelo da Revisão Cíclica. O nível máximo,  $M$ , para o qual são efectuadas as encomendas é determinado segundo a equação 2.51. Com a equação 2.52 é possível determinar a quantidade económica,  $Q^*$ , a encomendar e o intervalo entre revisões é determinado segundo a equação 2.53.

$$M = d \times (L + T) + s \quad (2.51)$$

$$Q^* = \sqrt{\frac{2 A d_{tr}}{C_2}} \quad (2.52)$$

$$T = \frac{Q^*}{d_{tr}} \quad (2.53)$$

$C_2$  é o custo de manter em *stock* uma unidade de artigo por unidade de tempo e  $d_{tr}$  é a procura média relativa ao tempo de revisão da política.

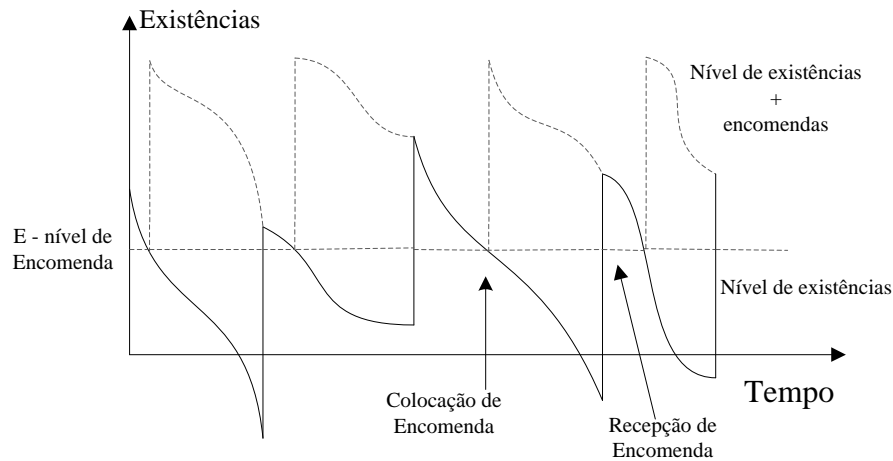
Efectuando este modelo uma optimização de *stocks* em que se estabelece um nível máximo para o qual se encomenda os produtos conclui-se que este modelo apenas é viável para casos cuja rotatividade dos produtos é elevada e o custo de armazenagem pouco relevante para a empresa.

Como forma de se perceber melhor este modelo está presente no Anexo D deste documento um exemplo prático relativo a um armazém de uma empresa de distribuição de uma marca de comida para gatos.

#### **2.2.2.2. Nível de Encomenda**

Neste modelo, sempre que o nível de *stock* é inferior a um determinado nível (nível de encomenda),  $E$ , é colocada uma encomenda de dimensão fixa pré-determinada. Através da figura 2.9 é possível observar-se um exemplo de como é efectuado o tratamento das encomendas e o tipo de evolução que se tem nos *stocks* considerando esta política de reaprovisionamento.

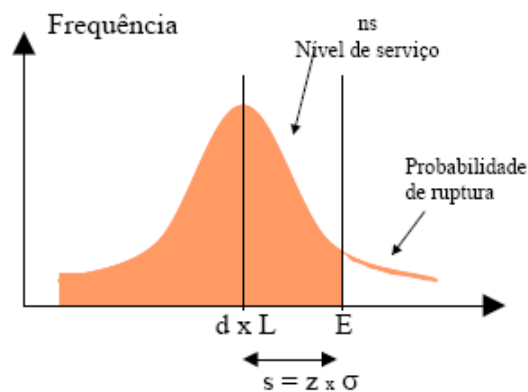
Observa-se aqui, tal como no modelo anterior, que entre os vários pontos de revisão há sempre a possibilidade de ocorrer rotura de *stock* devido ao tempo de reposição dos produtos ser desconhecido. De notar que, no caso deste modelo, os intervalos entre encomendas podem variar consoante a procura pois, tal como foi dito anteriormente, só quando se atinge o nível de encomenda é efectuada uma nova encomenda.



**Figura 2.9** – Nível de Encomenda: Tratamento de Encomendas

Devido a esta variação nos intervalos entre encomendas a grandeza  $T$  considerada no modelo de Revisão Cíclica não é considerada neste modelo para a determinação do *stock* de segurança.

Comece-se a análise de optimização dos *stocks* para este modelo pela determinação do respectivo *stock* de segurança,  $s$ . A figura 2.10 ilustra como analisar o *stock* de segurança já que este é determinado através da distribuição da procura do produto, tal como no modelo anterior.



**Figura 2.10** – Nível de Encomenda: Exemplo de Distribuição de Procura

O *stock* de segurança é calculado através do desvio padrão da procura,  $\sigma$ , e do acontecimento relativo ao nível de serviço ( $ns$ ) respectivo,  $z$ . Tal como no modelo anterior, a grandeza  $\sigma_{u.t.}$  refere-se ao desvio padrão relativo à unidade de tempo de reaprovisionamento.

$$\sigma = \sqrt{L} \times \sigma_{u.t.} \quad (2.54)$$

$$s = z \times \sigma \quad (2.55)$$

Está-se agora em condições de mostrar as relações necessárias à *optimização* dos *stocks* utilizando este modelo. O nível de encomenda ou ponto de encomenda, E, que determina que seja efectuado um novo pedido de encomenda, é determinado segundo a equação 2.56, sendo que d, se refere à procura média diária e L ao número de unidades de tempo para entrega de uma encomenda.

$$E = d \times L + s \quad (2.56)$$

A quantidade económica,  $Q^*$ , a encomendar é dada pela equação 2.57. De notar que  $d_{tr}$  representa a procura média relativa ao tempo de revisão da política e  $C_2$  se refere, tal como no caso anterior, ao custo de manter em *stock* uma unidade de artigo por unidade de tempo. T é o intervalo de tempo entre encomendas que se determina através da equação 2.58.

$$Q^* = \sqrt{\frac{2 A d_{tr}}{C_2}} \quad (2.57)$$

$$T = \frac{Q^*}{d_{tr}} \quad (2.58)$$

O modelo Nível de Encomenda pode ser usado tanto em casos de produtos de muita rotatividade como de pouca e, ao contrário do modelo anterior, é possível usar este modelo tanto em casos cujo custo de armazenagem é relevante ou não, ou seja, em que o espaço alocado e respectivos custos sejam ou não importantes.

Como forma de se perceber melhor este modelo, está presente no Anexo D deste documento um exemplo prático, tal como no modelo anterior relativo a um armazém de uma empresa de distribuição de uma marca de comida para gatos.

Alternativamente, e embora não tenha sido utilizado neste trabalho seria também possível utilizar o processo iterativo para este modelo, como se pode observar no Anexo E [Santos, 2005].

### 2.2.2.3. Política Mista

Este modelo é um misto dos dois modelos anteriores. O modelo baseia-se em realizar inspeções ao *stock* a intervalos fixos de tempo em que se o *stock* desceu abaixo de  $m$  (nível mínimo) é colocada um encomenda de dimensão variável que o eleva até ao nível  $M$  (nível máximo). Se o *stock* não desceu abaixo de  $s$ , não é colocada nenhuma encomenda.

### 2.2.2.4. Revisão Cíclica e Nível de Encomenda – Vantagens e Desvantagens

Na tabela 2.1 estão explicitadas as vantagens e desvantagens dos Modelos Revisão Cíclica e Nível de Encomenda.

	Revisão Cíclica	Nível de Encomenda
Vantagens	<ul style="list-style-type: none"> <li>▪ Encomendas colocadas a intervalos fixos de tempo.</li> <li>▪ Agregação de Encomendas.</li> </ul>	<ul style="list-style-type: none"> <li>▪ Encomendas de dimensão fixa.</li> <li>▪ Aplica-se tanto a casos de produção como de revenda.</li> </ul>
Desvantagens	<ul style="list-style-type: none"> <li>▪ Risco de rotura de stocks entre pontos de revisão.</li> <li>▪ Custos de armazenagem mais elevados assim como espaço alocado</li> </ul>	<ul style="list-style-type: none"> <li>▪ Necessário conhecimento contínuo do sistema.</li> </ul>

**Tabela 2.1** – Revisão Cíclica e Nível de Encomenda: Vantagens e Desvantagens

## 2.3. Análise ABC

Grande parte das empresas têm muitos produtos guardados em *stock*, podendo alguns desses produtos ser mais importantes para a empresa do que outros. Os mais importantes são aqueles que têm grande rotatividade, ou seja, aqueles que são muito usados ou vendidos, o que no caso de se esgotarem facilmente provocará descontentamento nos clientes que os queiram adquirir. Os menos importantes podem representar um grande valor de investimento, no caso de serem caros, assim como uma excessiva alocação de espaço de armazenagem. Reduzindo o

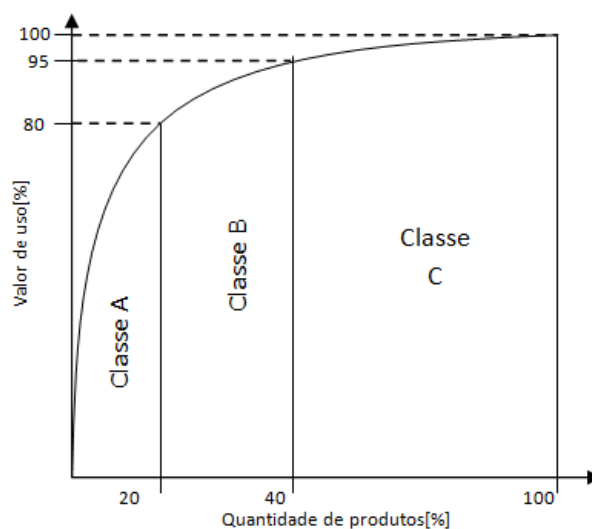


seu nível de *stock* poupa-se dinheiro e liberta-se espaço alocado para ter mais quantidade em *stock* dos produtos mais importantes. É assim importante fazer uma distinção entre os produtos da empresa ao nível do seu valor de uso. Um método que permite esta distinção ou classificação dos diversos produtos tidos em *stock*, designa-se por Análise ABC [Tanwari, 2000]. Este método pode ser implementado em vários tipos de negócios tais como matérias-primas, componentes de substituição, aquisição de diferentes tipos de peças para linhas de produção, produtos em geral ou produtos acabados, dependendo como está constituído o *stock* da empresa em questão.

A Análise ABC permite dividir um conjunto de produtos em 3 categorias diferentes, dependendo do seu valor de uso, ou seja, da sua importância ou significância para a empresa. Quanto maior for o valor de uso de um produto mais importante esse produto é para o negócio. Desta forma, esta análise tem impacto no custo geral de inventário ou *stock*. É assim importante definir, matematicamente, em que consiste o valor de uso, que se encontra expresso na equação 2.59.

$$\text{valor de uso} = \text{custo unitário do produto} \times \text{procura no período} \quad (2.59)$$

Na figura 2.11 é possível observar-se a chamada *Curva ABC*, ou seja, a evolução da relação entre a percentagem do valor de uso total e a percentagem da totalidade dos produtos.



**Figura 2.11** – Curva ABC

Neste tipo de análise, os níveis de significância, por ordem decrescente, correspondem, respectivamente, a três categorias designadas por A, B e C.

Os produtos da categoria A representam até 20% do total de produtos que, por sua vez, correspondem até 80% do valor de uso total dos produtos para a empresa. No que diz respeito a despesa, estes produtos são os produtos mais sensíveis e por isso os mais importantes para a empresa. Por esta razão são estes os produtos que precisam de controlo mais frequente e em relação aos quais deverão ser evitadas ao máximo roturas de *stock*.

Os produtos da categoria B representam 20% do total de produtos que, por sua vez, correspondem a uma gama de 15% do acumulado do valor de uso. Estes produtos requerem uma análise normal, ou seja, devem ser revistos ou analisados três a quatro vezes por ano.

Na categoria C os produtos representam 60% dos produtos tidos em *stock*. Estes produtos equivalem ainda a uma gama de cerca de 5% do acumulado do valor de uso dos produtos. Usualmente são produtos baratos ou de baixa rotatividade e por esta razão pode ser feita uma análise mais tolerante a falhas, sendo feita normalmente uma revisão por ano para estes produtos.

Como se verificará seguidamente, na explicação feita sobre o processo de classificação, para se poder efectuar a Análise ABC é necessário ter conhecimento sobre registos passados acerca dos produtos assim como previsões para os mesmos, dentro do período pretendido para análise. Desta forma é importante notar que também esta análise ABC é apenas um método de apoio à decisão para o responsável de gestão de *stocks* que permite que este tenha uma melhor percepção da qualidade do investimento que está a ser feito por parte da empresa ao nível do seu *stock*.

A seguir é feita uma breve descrição de todo este processo de análise ou classificação, ou seja, dos requisitos necessários, das grandezas que são necessárias calcular e como as calcular e ainda das conclusões que se tiram no final de todo este processo.

## **Descrição do Processo de Classificação**

Para cada produto é necessário saber o custo unitário e a procura desse produto no período a analisar, que deverá corresponder a um ano, calculando-se então o seu valor de uso. Os produtos são então ordenados por ordem decrescente de valor de uso e atribui-se um número de ordem (1, 2, ..., n). Seguidamente calcula-se a respectiva percentagem relativa ao total dos números de ordem atribuídos, o acumulado de valor de uso para cada um dos produtos e a percentagem do total relativa ao acumulado. Finalmente, com as percentagens anteriores, consegue-se a relação entre percentagem do total de produtos e percentagem do total do valor de uso dos produtos, relação esta que dita a classe a que pertencerá o produto.

Como forma de tornar esta descrição mais perceptível, no anexo F deste documento, encontra-se um exemplo prático de aplicação da Análise ABC.

# CAPÍTULO 3

## Trabalho Desenvolvido

---

Neste capítulo apresenta-se o trabalho desenvolvido ao longo desta tese. São apresentados os objectivos do trabalho, as opções tomadas ao longo do percurso de investigação e implementação. Com base no *software* implementado, designado por SAT (*Stock Analysis Tool*), apresentam-se os resultados obtidos.

### 3.1. Introdução

Tal como já foi referido na secção 1.2, os objectivos pretendidos centram-se na minimização dos custos de *stocks* na indústria de retalho. Previamente a essa análise dos *stocks*, pretende-se implementar um classificador que mostre a importância dos produtos tratados para o negócio.

O percurso de investigação deste trabalho começou com a escolha de uma entre várias áreas de negócio dos mercados de retalho. A área escolhida foi a indústria farmacêutica tendo sido fornecidos dados acerca de produtos farmacêuticos comercializados por uma farmácia em particular<sup>(\*)</sup>. Foi a partir destes dados que se implementou o classificador, referido no parágrafo anterior, assim como foi feita a optimização do *stock* dos produtos.

Com a análise efectuada ao *stock* é possível estabelecer uma comparação entre os resultados obtidos com a análise efectuada no SAT e com os parâmetros de *stock* utilizados no sistema de origem da farmácia, que se passará a designar por SF (Sistema da Farmácia).

<sup>(\*)</sup> Devido a compromissos de sigilo não se divulga o nome da farmácia em questão assim como eventuais nomes de produtos, tomados em exemplos apresentados ao longo desta dissertação.

Para além da implementação da Análise ABC (categorização dos produtos) e da análise estatística efectuada aos produtos é também efectuada uma pequena análise quanto à eventual sazonalidade dos produtos em determinados períodos do ano.

### **3.2. Análise ABC – Classificador Implementado**

Pretende-se com este classificador dar uma noção ao responsável pelo *stock* da farmácia da importância, ao nível financeiro, que cada um dos produtos comercializados tem para a mesma. Ou seja, quais os produtos com maior rotatividade e/ou mais dispendiosos para a farmácia (classe A), quais os que têm menor rotatividade e/ou são um pouco menos dispendiosos (classe B) que os primeiros e aqueles produtos que são financeiramente irrelevantes para a farmácia, ou seja, que têm escassa rotatividade ou que mesmo tendo alguma rotatividade são muito baratos (classe C). Outro propósito deste classificador é possibilitar a classificação de produtos novos para os quais não exista histórico.

#### **3.2.1. Implementação**

Tal como foi explicado na secção 2.3, a grandeza que afecta directamente esta classificação é o valor de uso do produto. O valor de uso é uma grandeza que é determinada através de dois atributos específicos relativos ao produto em causa: o custo de aquisição e a procura média do produto ao longo do ano em estudo. Com base nestes dois atributos e relativamente ao ano 2006, que era o único ano na base de dados para o qual se tinha os movimentos ao longo de todo o ano, fez-se a extracção dos produtos a analisar no decurso desta dissertação. Desta extracção resultaram 2636 produtos que foram utilizados em todas as fases de desenvolvimento deste trabalho.

O classificador concebido é um classificador baseado em árvores de decisão [Fonseca, 1994].

As árvores de decisão são compostas por nós e respectivos ramos. Dos quais se extraem as regras ou condições que ditam a classificação pretendida. De forma a extrair as regras de classificação utilizou-se um *software* específico, designado por CART.

Para se efectuar esta classificação é necessário dois conjuntos de dados: um conjunto de treino e um conjunto de teste. O conjunto de treino é o conjunto que tem o maior número possível de dados, que deverão ter uma classificação atribuída *a priori*, normalmente efectuada por um perito. Este conjunto é o conjunto de dados utilizado no CART para extracção de regras de classificação. O conjunto de teste são os dados utilizados para validação das regras de classificação geradas pelo CART.

Foi então aplicado, aos 2636 produtos, o processo de classificação mencionado na secção 2.3. Na tabela 3.1 é possível observar 15 dos 2636 produtos *treinados*. De notar que, na tabela 3.1, os valores atribuídos às classes foi 1, 2 e 3 que correspondem, respectivamente, às classes A, B e C. A numeração dada às classes deveu-se unicamente à utilização do *software* CART cuja variável da classe tem de ser numérica.

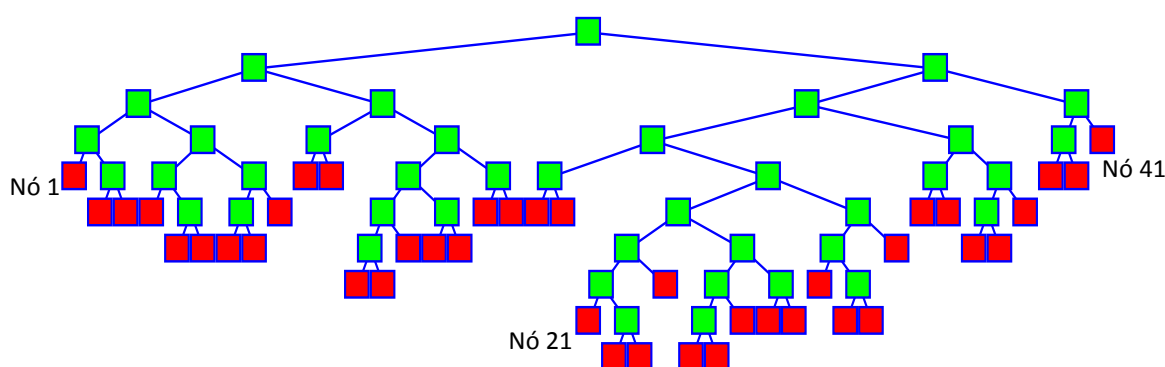
ID_PRODUTO	COST_ACQ	PROCURA	USE_VALUE	ORDER_NR	PERC_ORDER	USE_VAL_CUMUL	PERC_USE_VAL_CUMUL	CLASS
A1	16.397	382	6263.65	1	0.038	6263.654	1.582	1
A2	39.691	149	5913.96	2	0.076	12177.613	3.077	1
A3	32.851	177	5814.63	3	0.114	17992.240	4.545	1
A4	17.86361904	295	5269.77	4	0.152	23262.008	5.877	1
A5	32.851	152	4993.35	5	0.190	28255.360	7.138	1
B1	3.9045	41	160.08	527	19.992	319837.460	80.804	2
B2	3.553	45	159.89	528	20.030	319997.345	80.844	2
B3	10.5355	15	158.03	529	20.068	320155.377	80.844	2
B4	11.286	14	158.00	530	20.106	320313.381	80.924	2
B5	4.921	32	157.47	531	20.144	320470.853	80.964	2
C1	5.6905	6	34.14	1292	49.014	379259.592	95.817	3
C2	11.381	3	34.14	1293	49.052	379293.735	95.826	3
C3	6.82	5	34.10	1294	49.090	379327.835	95.834	3
C4	11.34666667	3	34.04	1295	49.127	379361.875	95.843	3
C5	3.781	9	34.03	1296	49.165	379395.904	95.851	3

**Tabela 3.1** – Alguns dados trabalhados pertencentes ao conjunto de treino/teste

Dos 2636 produtos analisados, 2/3 formam o conjunto de treino e 1/3 forma o conjunto de teste.

### 3.2.2. Apresentação de Resultados

Foi então gerada uma árvore de decisão com 41 nós terminais, como se pode observar pela figura 3.1. Destes 41 nós terminais extraem-se as regras que formam o classificador a usar no SAT.



**Figura 3.1** – Árvore de Decisão Gerada

As regras do classificador (ver Anexo G) foram então testadas no conjunto de teste e posteriormente para todos os 2636 produtos analisados. A matriz de confusão (tabela 3.2) representa os resultados do teste do classificador [Fonseca, 1994]. Como se pode observar pela tabela 3.2, dos 2636 produtos analisados, as regras de classificação falharam na análise de apenas 54 desses produtos correspondendo a uma taxa de sucesso de 98%. Verifica-se também que nenhum produto de classe A foi classificado como classe C.

		Classe Real		
		A	B	C
Classe Estimada	A	522	26	1
	B	4	735	19
	C	0	4	1325

**Tabela 3.2** – Matriz de Confusão

Os erros de classificação não são graves pois se a situação anterior se verificasse, os produtos que são muito importantes para o negócio e que devem ser alvo de grande atenção ao nível das suas reposições em *stock* poderiam sofrer inúmeras quebras de *stock* ao longo do ano comercial. Deste modo, o negócio poderia ter prejuízos em relação a esses produtos, que podem significar perda de clientes.

### 3.3. Análise de *Stocks* – Modelo Nível de Encomenda

Tal como se referiu na secção 2.2 deste documento, foram analisados, previamente à implementação deste trabalho, diferentes modelos de análise de *stocks*, nomeadamente, modelos determinísticos e estocásticos. Sendo os modelos determinísticos dependentes da procura média esperada de um produto e devendo esta ser previamente conhecida concluiu-se rapidamente que estes modelos não eram adequados ao caso da área de negócio em causa. A necessidade que uma pessoa possa ter por um medicamento é imprevisível pois depende fundamentalmente do seu estado de saúde que pode variar de ano para ano. Foi esta a razão pela qual se concluiu que estes modelos determinísticos não são de aplicação directa no caso do negócio em causa.

Após o estudo dos modelos anteriores passou-se ao estudo dos modelos estocásticos apresentados na secção 2.2, ou seja, o Modelo Nível de Encomenda e o Modelo de Revisão Cíclica.

Após a tentativa de implementação de ambos os modelos concluiu-se que o modelo de Revisão Cíclica não era adequado ao problema em estudo. Esta não adequação do modelo deveu-se ao facto deste modelo se concentrar em *optimizar* o *stock* máximo dos produtos, ou dito de outra forma, a quantidade encomendada era determinada de forma a que o nível de *stock* atingisse um pré-determinado valor máximo ao fim de um determinado tempo. Relembrando que o objectivo deste trabalho é minimizar os custos de *stock* torna-se



necessário ter uma quantidade mínima de *stock* objectivo este que não é possível com o modelo Revisão Cíclica.

Concentrando-se o Modelo Nível de Encomenda na *optimização* do *stock* mínimo dos produtos foi então este o modelo adoptado.

### 3.3.1. Implementação

A optimização de *stocks* efectuada neste trabalho, utilizando o Modelo Nível de Encomenda, concentra-se no estudo e *optimização* dos *stocks* mínimos e máximos adoptados pelo SF (Sistema da Farmácia), especificamente o *stock* mínimo. De notar que no SF as encomendas são efectuadas sempre que a quantidade em *stock* de cada produto atinge o seu valor mínimo.

Após se ter todos os produtos classificados guardados em base de dados, mais concretamente, na tabela PRODUCT (ver anexo A), é possível efectuar esta optimização do *stock* de um ou de todos os produtos em simultâneo.

Para tal, é necessário calcular algumas grandezas, tal como foi descrito na secção 2.2.2.2. Começou-se então por extrair o custo de aquisição dos produtos, seguido do tempo médio de reposição dos mesmos.

O tempo médio de reposição dos produtos é uma das principais variáveis que influenciam esta análise de *stocks*. Torna-se portanto muito importante utilizar da melhor forma esta grandeza. Segundo o modelo e como se pode ver pelo exemplo tido no anexo D, o tempo médio de reposição a utilizar deve ser em fracção de dias, ou seja, se um produto tiver um tempo médio de reposição de 12 horas este corresponde a 0,5 dias. O modo como esta grandeza é calculada baseia-se em fazer o valor médio da diferença entre a data de recepção da encomenda e a data de envio da mesma para o fornecedor.

O nível de serviço dos produtos utilizado nesta análise foi fixo. Considerou-se que a SAT e o seu sistema de análise dos *stocks* tinha um nível de serviço de 95%, ou seja, apenas 5% da procura pode não ser satisfeita.

A procura média de cada produto foi determinada com base nas amostras tidas na base de dados fornecida. Determinou-se esta média assim como o desvio padrão associado a esta que também é necessário. Para cada produto são retiradas todas as vendas verificadas e guardam-se na tabela AUXILIARDEMAND (ver Anexo A), a partir da qual se efectua o cálculo do desvio padrão da procura pela equação 2.48, presente na secção 2.2.2.

Outro custo importante para o cálculo dos parâmetros de optimização é o custo de processamento de uma encomenda que se considerou independente do produto ou do número de produtos contidos em cada encomenda. Quando qualquer encomenda tem que ser processada ou recepcionada um dos colaboradores da farmácia fica alocado a esse serviço por um determinado tempo. Para esta análise considerou-se que o funcionário da farmácia gasta 15 minutos do seu tempo de trabalho para processar ou recepcionar cada encomenda. Sabendo o valor típico de ordenado de um funcionário de farmácia que trata das encomendas, extrai-se quanto este ganha aproximadamente por hora, considerando que o mês tem 4 semanas, que se trabalha 6 dias por semana e 9 horas por dia retira-se então um quarto deste valor, como se pode ver pela equação 3.1.

$$\text{Custo de Processamento de uma encomenda} = \left( \frac{\text{ordenado mensal}}{9 \text{ horas} \times 6 \text{ dias} \times 4 \text{ semanas}} \right) \times 0.25 \quad (3.1)$$

O custo de processamento das encomendas guarda-se na tabela PHARMACY (ver Anexo A).

Como se pode observar pela equação 2.57, a quantidade económica a encomendar depende de  $C_2$ , o custo de manter em *stock* uma unidade de produto. Este custo determina-se com base no custo de aquisição do produto considerando-se como sendo cerca de 20% desse valor.

Seguindo as equações apresentadas na secção 2.2.2.2 calculou-se os parâmetros de *optimização* para os produtos pretendidos.

Encomendando-se apenas quando as unidades em *stock* atingem o *stock* mínimo então o novo *stock* mínimo (equação 3.2), não é dado pelo Ponto de Encomenda, E, tal como o modelo Nível de Encomenda indica. Já o valor do *stock* máximo é o valor do *stock* mínimo adicionado da quantidade económica a encomendar, Q, tal como a equação 3.3 o mostra.

$$\text{stock mínimo} = E - 1 \quad (3.2)$$

$$\text{stock máximo} = E - 1 + Q \quad (3.3)$$

### 3.3.2. Apresentação e Comparação de Resultados

A optimização por Nível de Encomenda pode ser feita para um único produto escolhido pelo utilizador ou, em simultâneo, para todos os produtos classificados anteriormente. O processo é igual para qualquer uma das duas opções. Sendo assim, mostra-se nesta secção a análise efectuada a um produto e comparam-se os *stocks* mínimo e máximo anteriormente usados e os calculados com o *software* SAT.

Mostra-se também nesta secção a implementação e os resultados obtidos pela funcionalidade *Simulação dos Stocks*, a partir da qual é possível comparar os comportamentos dos *stocks* ao longo do período analisado usando os *stocks* mínimo e máximo dos sistemas SF e SAT, respectivamente. Daqui é possível observar, ao longo dos 12 meses do ano, quanto se pouparia considerando um produto em particular ou considerando todos os produtos.

#### 3.3.2.1. Resultado da Implementação do Modelo Nível de Encomenda – Um Exemplo

Para este exemplo de demonstração do resultado de *optimização* de um produto considere-se um produto hipotético, que se passa a designar por produto X.

Independentemente do produto e considerando que o funcionário que trata da recepção ou processamento das encomendas auferir um ordenado mensal de 1000€ então o custo de processamento de encomendas é, aproximadamente, 1,16€.

Sabendo que o produto X, de classe A, tem um custo de aquisição de 16,44€ e um tempo médio de reposição de 11 horas e 45 minutos (0,489 dias) e que a procura média registada no ano analisado foi de, aproximadamente, 1,274 unidades por dia (a esta média da procura está associado um desvio padrão de 1,333 unidades). Vamos então proceder à sua análise.

$$Q^* = \sqrt{\frac{2 \times 1,16 \times 1,274}{0,2 \times 16,44}} \cong 1$$

$$T = \sqrt{\frac{2 \times 1,16}{1,274 \times 0,2 \times 16,44}} \cong 1 \text{ dia}$$

$$\text{Intervalo entre encomendas} = \frac{365}{T} \cong 365 \text{ dias}$$

$$\sigma = \sqrt{L} \times \sigma_{u.t.} = 0.932$$

$$s = z \times \sigma = 1.65 \times 0.932 \cong 2$$

$$E = \text{procura média} \times L + s = 3$$

$$\text{stock mínimo} = 2$$

$$\text{stock máximo} = 3$$

O responsável pelos *stocks* da farmácia pode assim concluir que será efectuada e colocada uma encomenda de 1 unidade sempre que o *stock* seja igual ou inferior a 2 unidades. O *stock* de segurança será de 2 unidades de produto, estando previstas 365 encomendas ao longo do ano, ou seja, tem-se um intervalo entre encomendas de, aproximadamente, 1 dia. Os *stocks* mínimo e máximo para o produto XPTO são, respectivamente, 2 e 3 unidades.

De notar que no sistema SF, os *stocks* mínimo e máximo do mesmo produto são, respectivamente, 3 e 4 unidades.

### 3.3.2.2. Simulação de Ganhos Obtidos

Tal como foi referido anteriormente, o objectivo deste trabalho é fornecer uma ferramenta que permita a minimização de custos de *stocks* sem causar um impacto significativo na qualidade

de serviço. Deste modo foi criada uma funcionalidade na SAT, designada por *Simulação de Stocks*, com a qual é possível ter noção acerca dos ganhos financeiros que, eventualmente, se teria utilizando os parâmetros de *stock otimizados* por esta ferramenta. Nesta secção explica-se como foi implementada a funcionalidade, as noções necessárias para se executar este processo e que resultados se obtêm.

A simulação dos ganhos obtidos é dividida em duas fases distintas: simulação dos *stocks*, ao longo do período pretendido, no sistema SF e na SAT e conclusões que se tiram da simulação anterior, ou seja, o cálculo dos ganhos e perdas nos dois sistemas. De notar que o intervalo de tempo a analisar é escolhido, exclusivamente, pelo utilizador.

### **Simulação dos *stocks***

No caso do sistema SF são extraídos todos os movimentos ao longo do período de análise especificado pelo utilizador. Os movimentos extraídos podem ser relacionados com saídas do sistema (vendas com receita, vendas sem receita, vendas suspensas, devolução a fornecedores, entre outros) ou entradas em sistema (entrega de encomenda, regularização de devolução, alterações manuais, anulação de venda, entre outros). É ainda possível simular vendas não satisfeitas por parte da farmácia utilizando esta funcionalidade, processo que é explicado mais adiante.

Para o sistema SAT é feita, na realidade, a simulação dos *stocks*, ou seja, dos dados fornecidos pela farmácia apenas se consideram as saídas. Das entradas em sistema apenas se extraem os casos de anulações de vendas ao cliente enquanto que as encomendas são determinadas pelo SAT. Quanto a eventuais vendas não satisfeitas, estas são previstas, tanto para o SAT como para o SF, com base na procura média esperada.

A estimação das encomendas para o sistema SAT é feita com base no tempo médio de reposição de cada produto. Ou seja, assim que o nível actual de *stock*, após um movimento, seja igual ou inferior ao *stock* mínimo é disparada uma encomenda, para o produto em análise,

consoante o tempo médio de reposição e a quantidade a encomendar obtida no processo de *optimização dos stocks*. Como não se tem acesso ao planeamento horário da farmácia em cada dia então todas as encomendas previstas cuja hora de entrega seja anterior ou posterior ao horário considerado para a farmácia (9h às 19h), são consideradas como recepcionadas em horas e/ou dias específicos. Ou seja, se a previsão da entrega for entre as 0h00 e as 9h então a encomenda é dada como entregue às 9h e se a previsão da entrega for entre as 19h e as 24h então a encomenda é dada como entregue no dia seguinte às 9h.

A estimação das vendas não satisfeitas ou de procura esperada não satisfeita é efectuada com base na procura média diária determinada anteriormente. Mais concretamente, o inverso da média da procura multiplicado por 86400<sup>(\*)</sup> segundos resulta no tempo esperado para a próxima procura em segundos.

Este tempo esperado é adicionado à data do movimento actual obtendo-se assim a data para a próxima procura esperada. Este processo é feito para ambos os sistemas, SF e SAT.

Na simulação dos *stocks* são ainda devolvidas as roturas verificadas/esperadas ao longo do período de análise seleccionado.

### **Conclusões a tirar da simulação anterior**

As conclusões que se tiram após a simulação dos *stocks* dizem respeito aos ganhos que se obtêm utilizando os parâmetros de *stock optimizados* pelo SAT. É possível efectuar a simulação para um ou para vários produtos em simultâneo e ambas as funcionalidades seguem o mesmo processo de análise.

No caso da simulação de um único produto são apresentados resultados quer tenha ganhos ou perdas. No caso da simulação de todos os produtos em simultâneo, no final apenas são apresentados resultados dos produtos que apresentem ganhos financeiros para a empresa.

(\*) 86400 segundos corresponde a 1 dia.

Para o SF e SAT, é necessário extrair o *stock* médio diário positivo e negativo do produto.

É ainda necessário ter conhecimento acerca da percentagem do valor médio de *stock* da farmácia a que corresponde o custo de armazenagem. Esta grandeza é relevante na medida em que ter em *stock* uma ou mais unidades de qualquer artigo tem um custo associado designado por custo de armazenagem. Para o cálculo desta grandeza foi efectuada uma abordagem simplista considerando-se apenas algumas das despesas possíveis que influenciam este custo. Deste modo, para o custo de armazenagem (equação 3.7) considerou-se a despesa tida com a limpeza do armazém, a despesa com a arrumação do armazém, a renda mensal correspondente à percentagem da área da farmácia a que o armazém corresponde e a factura mensal da electricidade.

$$\text{custo de limpeza mensal} = \frac{\text{rendimento}}{\text{por hora}} \times \frac{\text{horas por}}{\text{semana}} \times 4 \times \frac{\text{área de}}{\text{armazém}} \times \frac{100}{100} \quad (3.4)$$

$$\text{custo do arrendamento mensal} = \frac{\text{renda}}{\text{mensal}} \times \frac{\text{área de}}{\text{armazém}} \times \frac{100}{100} \quad (3.5)$$

$$\text{custo de arrumação mensal} = \frac{\text{tempo de}}{\text{arrumação}} \times \frac{\frac{\text{ordenado}}{\text{mensal}}}{\frac{\text{n}^\circ \text{ de dias de}}{\text{trabalho por mês}} \times \frac{\text{n}^\circ \text{ de dias de}}{\text{trabalho por mês}} \times \frac{100}{100}} \quad (3.6)$$

$$\text{custo de armazenagem} = \text{custo de limpeza mensal} + \text{custo de arrendamento mensal} + \text{custo de arrumação mensal} + \text{factura mensal de electricidade} \quad (3.7)$$

Dividindo o custo de armazenagem pelo valor total de *stock* da farmácia (obtido com base na tabela PHARMACY apresentada no Anexo A) obtém-se a percentagem pretendida.

Relativamente ao ganho que se obtém para cada produto, este é obtido pela diferença entre os *stocks* médios positivos do SF e do SAT. Numa segunda aproximação (equação 3.8) sendo esta diferença positiva então o actual sistema tem produto em excesso e sabendo que os *stocks* de uma qualquer farmácia valem milhares de euros então a farmácia poderia utilizar esse dinheiro investido em excesso de produtos para efectuar investimento em aplicações

financeiras. Considerando-se uma taxa de juro de 5% aplicada ao ano obteriam-se esses juros recebidos correspondentes a cada embalagem em excesso assim como em relação ao armazém que poderia ser mais pequeno e portanto mais económico do que o utilizado.

$$\text{ganho} = \left( \frac{n^{\circ} \text{ de dias}}{365 \text{ dias}} \right) \times \left( \frac{\text{stock médio}}{\text{positivo do SF}} - \frac{\text{stock médio}}{\text{positivo do SAT}} \right) \times \left( 5\% + \frac{\text{percentagem do}}{\text{valor de stock}} \right) \times \frac{\text{custo de}}{\text{aquisição}} \quad (3.8)$$

No caso das perdas, ou seja, quando se tem roturas devido a não vendas, que se possam verificar para cada produto, são apenas necessários os *stocks* médios negativos do SF e do SAT, a percentagem dos clientes não satisfeitos que se possam perder (neste caso apresentam-se resultados considerando 15% e 30%) e o lucro que se perde quando uma venda não é satisfeita. Este lucro é a diferença entre o preço de venda ao público e o custo de aquisição por parte da farmácia aos fornecedores.

A equação 3.9 mostra o cálculo das perdas considerando que 15% dos clientes não satisfeitos já não voltam para comprar o produto. Para o caso de perda de 30 % dos clientes não satisfeitos o cálculo é análogo ao caso anterior.

$$\text{perdas}_{[15\%]} = \text{Abs} \left( \frac{\text{stock médio}}{\text{negativo do SF}} - \frac{\text{stock médio}}{\text{negativo do SAT}} \right) \times 15\% \times \text{lucro} \quad (3.9)$$

Finalmente obtêm-se os ganhos totais pretendidos, para cada produto, como conclusão da simulação efectuada considerando 15% de perda de clientes não satisfeitos (equação 3.10) e 30% de perda de clientes não satisfeitos, sendo este análogo à equação 3.10.

$$\text{Ganho Total}_{[15\%]} = \text{ganho} - \text{perdas}_{[15\%]} \quad (3.10)$$

Os resultados, podem ser mostrados de forma diferente consoante a funcionalidade pela qual o utilizador optou, simulação de um produto ou de todos os produtos em simultâneo.

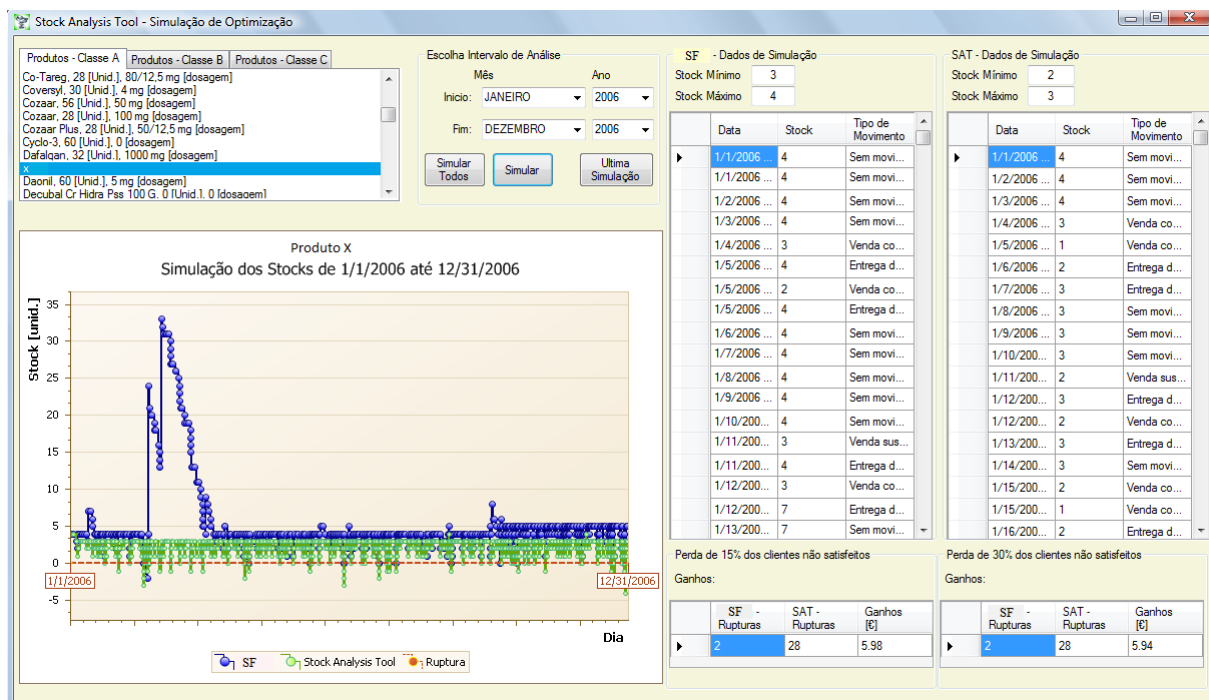


## Simulação de um produto com dados históricos

Para apresentação de resultados desta funcionalidade foi escolhido o mesmo produto utilizado na secção 3.3.2.1, que se designou por X.

Como se pode observar pela figura 3.2, esta simulação mostra a evolução do *stock* do produto X para o período temporal de um ano. A evolução deste *stock* é feita em termos gráficos assim como em detalhe (data do movimento, quantidade tida em *stock* após esse movimento e o tipo do movimento) através de duas tabelas, utilizando os parâmetros do sistema SF e os do SAT, respectivamente. Como forma de comparação entre os dois sistemas, mostra-se ainda os *stocks* mínimo e máximo utilizados por ambos os sistemas.

Finalmente, mostra-se o pretendido para esta funcionalidade, ou seja, os ganhos ou perdas que se obtêm utilizando os novos parâmetros de *optimização* calculados com a ferramenta SAT. São mostrados dois resultados diferentes considerando-se que 15% e 30% dos clientes não satisfeitos não voltam ao estabelecimento para comprar o produto, respectivamente.



**Figura 3.2 – Simulação de ganhos para o produto X**

Os valores dos *stocks* mínimo e máximo do produto X diferem relativamente ao sistema utilizado:

- *stock* mínimo de 3 unidades e *stock* máximo de 4 unidades utilizando o sistema SF;
- *stock* mínimo de 2 unidades e *stock* máximo de 3 unidades utilizando o sistema SAT.

O sistema SAT evidencia um *stock* mínimo inferior ao do SF e assim tem-se a vantagem de se poder poupar financeiramente nas encomendas de unidades deste produto assim como ter menos espaço de armazenagem ocupado.

Para além da vantagem apresentada tem-se também desvantagens ao reduzir o valor de *stock* mínimo do produto. Pois desta forma aumenta-se a probabilidade de ocorrência de roturas de *stock* dependendo de picos de procura do produto, ou seja, aumenta a probabilidade de se perder algumas vendas.

Assim, apresentam-se os ganhos ou perdas que se obtêm utilizando os parâmetros da SAT, assim como o número de roturas ocorridas/estimadas utilizando os dois sistemas em comparação.

Observando a figura 3.2, conclui-se que no sistema SF ocorrem 2 roturas enquanto que na SAT se verificam 28 roturas.

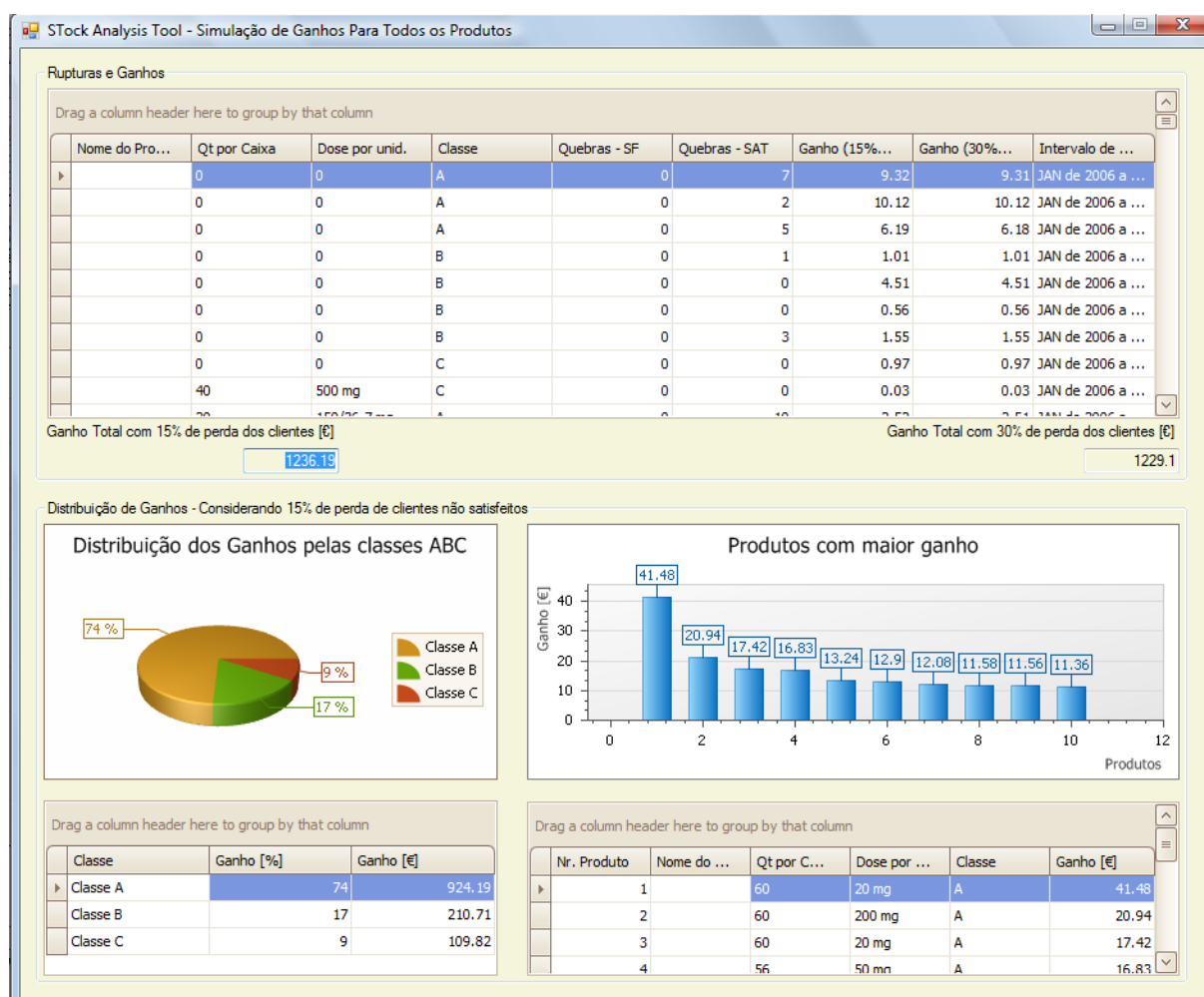
Considerando que 15% dos clientes não satisfeitos não voltam para comprar o produto, obtém-se um ganho de, aproximadamente, 5,98€. Considerando que 30% dos clientes não satisfeitos não voltam para comprar o produto, obtém-se um ganho de, aproximadamente, 5,94€.

Em qualquer um dos casos anteriores, os ganhos verificados devem-se ao facto de em dois períodos diferentes o sistema implementado, SF, ter feito encomendas de número elevado de unidades de produto, tendo tido no máximo 34 produtos em armazém. Assim e durante muito tempo a farmácia teve demasiadas unidades em armazém enquanto que com o sistema SAT o

valor máximo que se teve em *stock* neste período foi de 4 unidades, sendo que com este último sistema se teria poupado ao nível das quantidades encomendadas e mantidas em *stock*.

### Simulação de todos os produtos em simultâneo

A simulação de todos os produtos em simultâneo, como se pode observar pela figura 3.3, mostra a lista de produtos para os quais se conseguiu ter ganhos financeiros, ao fim do período escolhido, e respectivo valor do ganho total para todos os produtos. Mostra-se ainda a distribuição dos ganhos pelas classes de produtos e os dez produtos com maior ganho. Estes resultados dizem respeito a uma simulação para um período de um ano.



**Figura 3.3** – Simulação de ganhos para todos os produtos

Conclui-se então que, para os produtos com ganhos positivos, obtêm-se ganhos globais de 1236,19€ considerando perda de 15% dos clientes não satisfeitos e 1229,10€ quando se considera que 30% dos clientes não satisfeitos são perdidos.

Quanto à distribuição dos ganhos (considerando que 15% dos clientes não satisfeitos são perdidos), a classe A, tal como esperado, é a classe com a qual se consegue maior ganho, de 924,19€, seguida da classe B com ganhos de 210,71€ e por fim a classe C com ganhos de 109,82€. Quanto aos produtos com maior ganho (ver gráfico e tabela presentes no canto inferior direito da figura 3.3), o 10º produto com maior ganho é de classe C com um ganho de 11,36€ e os nove primeiros produtos são de classe A. O maior ganho obtido é de 41,48€.

Destas constatações é possível também confirmar, uma vez mais, que os produtos cuja classificação é A são os mais importantes para o negócio em análise.

### **3.4. Sazonalidade de Produtos**

A sazonalidade dos produtos é também importante para a gestão de *stocks* pois existem produtos cuja procura se verifica maioritariamente em períodos específicos do ano, podendo ser nula ou muito reduzida nos restantes períodos. Com esta funcionalidade, apenas se verifica que produtos são considerados sazonais no período definido pelo utilizador.

#### **3.4.1. Implementação**

O estudo da sazonalidade é possível de duas formas diferentes:

1. O utilizador escolhe uma ou duas das estações do ano apresentadas;
2. O utilizador escolhe no máximo 6 meses dos meses disponíveis na janela apresentada.

Em qualquer um dos casos anteriores, é ainda necessário que o utilizador insira na caixa de texto apresentada a percentagem das vendas anuais para a qual o sistema SAT deve considerar

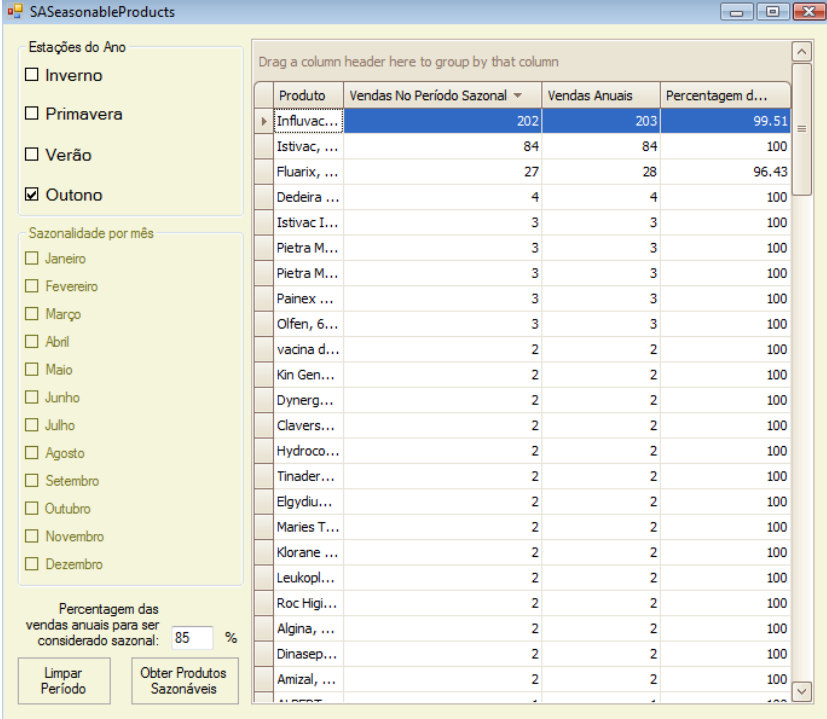
o produto como sazonal. Após a escolha do tipo de sazonalidade e percentagem de vendas a considerar, são efectuadas duas inspecções à base de dados com as quais se extraem as vendas no período escolhido (vendas\_no\_período) e no ano todo (vendas\_totais), respectivamente. Se o quociente destas duas grandezas for superior à percentagem inserida pelo utilizador então o produto é sazonal no período escolhido.

### 3.4.2. Apresentação de Resultados

Mostram-se então dois resultados utilizando cada uma das opções de período a estudar. Foi considerado um valor de 85% para percentagem de vendas anuais, a partir da qual o produto é considerado sazonal.

#### 3.4.2.1. Sazonalidade por Estações do Ano

Neste caso, para demonstração de resultados considerou-se a estação do outono. A figura 3.4 mostra os resultados obtidos.



SASeasonableProducts

Estações do Ano

☐ Inverno

☐ Primavera

☐ Verão

☒ Outono

Sazonalidade por mês

☐ Janeiro

☐ Fevereiro

☐ Março

☐ Abril

☐ Maio

☐ Junho

☐ Julho

☐ Agosto

☐ Setembro

☐ Outubro

☐ Novembro

☐ Dezembro

Percentagem das vendas anuais para ser considerado sazonal: 85 %

Limpar Período

Obter Produtos Sazonáveis

Drag a column header here to group by that column

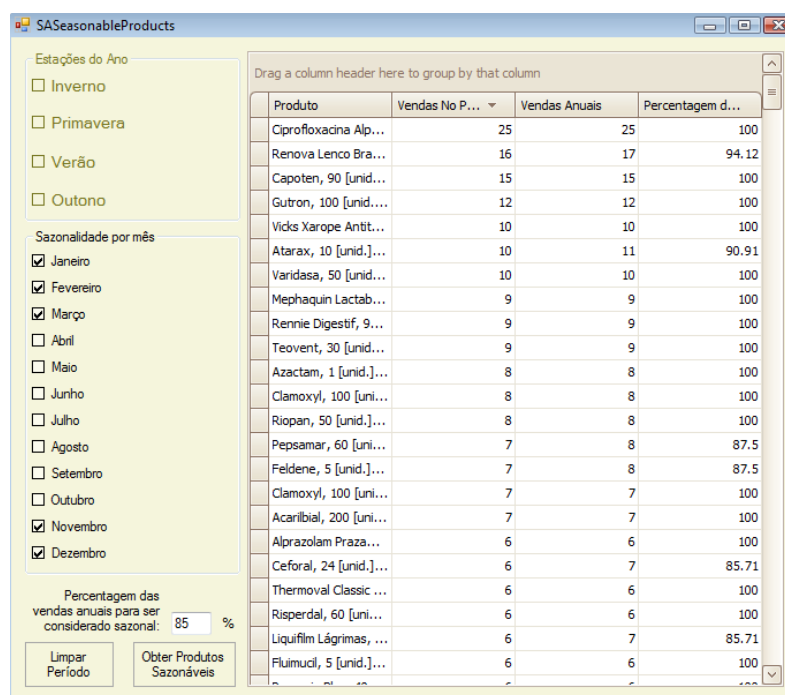
Produto	Vendas No Período Sazonal	Vendas Anuais	Percentagem d...
Influvac...	202	203	99.51
Istivac, ...	84	84	100
Fluarix, ...	27	28	96.43
Dedeira ...	4	4	100
Istivac I...	3	3	100
Pietra M...	3	3	100
Pietra M...	3	3	100
Painex ...	3	3	100
Ofen, 6...	3	3	100
vacina d...	2	2	100
Kin Gen...	2	2	100
Dynerg...	2	2	100
Clavers...	2	2	100
Hydroco...	2	2	100
Tinader...	2	2	100
Elgydiu...	2	2	100
Maries T...	2	2	100
Klorane ...	2	2	100
Leukopl...	2	2	100
Roc Higi...	2	2	100
Algina, ...	2	2	100
Dinasep...	2	2	100
Amizal, ...	2	2	100

**Figura 3.4 – Sazonalidade por Estações do Ano – Outono**

A estação do ano escolhida, o Outono, diz-se que é uma estação de alergias e começo de aparecimento de algumas doenças sazonais do frio como, por exemplo, a gripe. Dos resultados, nota-se que os três primeiros produtos que aparecem, cujo número de vendas foi maior, são três vacinas contra a gripe sazonal e que, na realidade, se tomam normalmente no Outono.

### 3.4.2.2. Sazonalidade por Mês

Neste caso, para demonstração de resultados considerou-se os meses de Janeiro, Fevereiro, Março, Novembro e Dezembro. A figura 3.5 mostra os resultados referidos.



**Figura 3.5 – Sazonalidade por mês**

Os 5 meses do ano escolhidos são meses cujas temperaturas são as mais baixas ao longo do ano, ou seja, são meses nos quais aparecem mais gripes e constipações, entre outras doenças ou infecções habituais neste período. Dos resultados presentes na figura 3.5 é perceptível a presença destas doenças com a presença de antibióticos/antivirais (por exemplo, Clamoxil 100), anti-inflamatórios (por exemplo: Varidasa). Outros medicamentos para tratamento de inflamações/infecções respiratórias (por exemplo: Vicks Xarope, Fluimucil) também são detectados o que era também esperado.

### 3.5. Software Desenvolvido (*Stock Analysis Tool*)

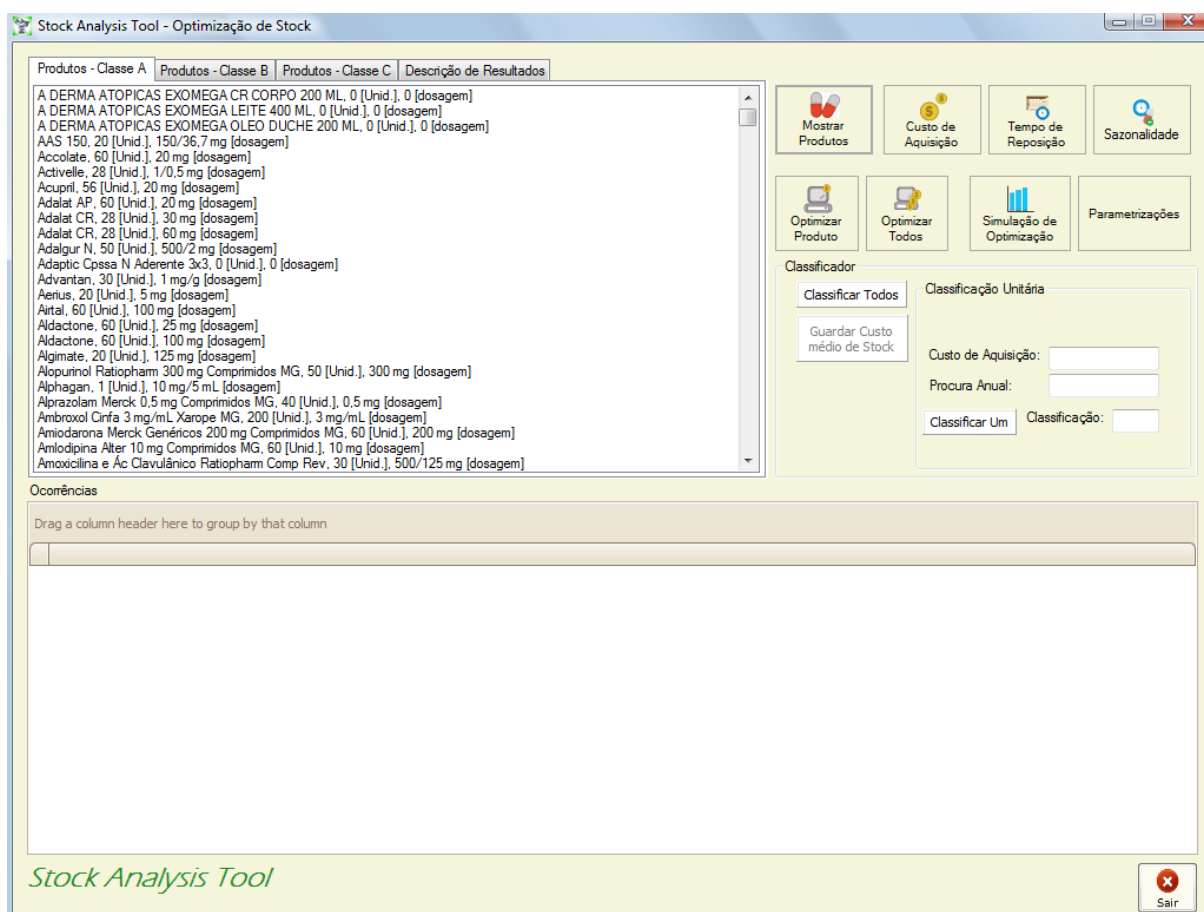
Terminada a investigação teórica necessária ao desenvolvimento desta tese foi implementada a aplicação que serve para demonstração dos resultados da investigação, referidos nas secções anteriores. Esta aplicação designa-se por *Stock Analysis Tool*, ou abreviadamente, SAT.

Várias tecnologias foram utilizadas na implementação desta aplicação. A plataforma de programação utilizada foi o *Microsoft Visual Studio 2005 (.Net)* por meio da linguagem de programação C# que é uma linguagem orientada por objectos<sup>(\*)</sup>. As classes implementadas correspondem também, no caso, desta aplicação, às funcionalidades disponíveis. Destas classes, tem-se apenas uma classe mais geral, designada por *StockOpt* onde se desempenham todas as acções, cálculos a efectuar e ligações a *web services*. As restantes classes dizem respeito apenas e só às interfaces gráficas (*forms*) implementadas onde são mostrados os resultados de cada funcionalidade.

Embora já tenham sido referidas nas secções anteriores, as funcionalidades/*forms* disponíveis nesta aplicação são mostrados e descritos seguidamente.

Tem-se um *form* inicial, como se pode observar pela figura 3.6, onde é possível obter diversos resultados, tais como, o custo de aquisição de um produto, o tempo médio de reposição de um produto, efectuar a classificação dos produtos e optimização de um produto escolhido pelo utilizador assim como de todos os produtos em simultâneo. É ainda possível testar a classificação de um produto novo mediante a inserção dos valores da procura anual esperada e do custo de aquisição do produto.

(\*) Os objectos de uma arquitectura de software são definidos através da implementação de uma ou mais classes. Estas classes é onde se define os métodos associados ao objecto, ou seja, que comportamentos pode este tomar. Define-se ainda os atributos ou variáveis associadas ao objecto, ou seja, os estados que os objectos podem tomar [JavaTutorials, 2010].



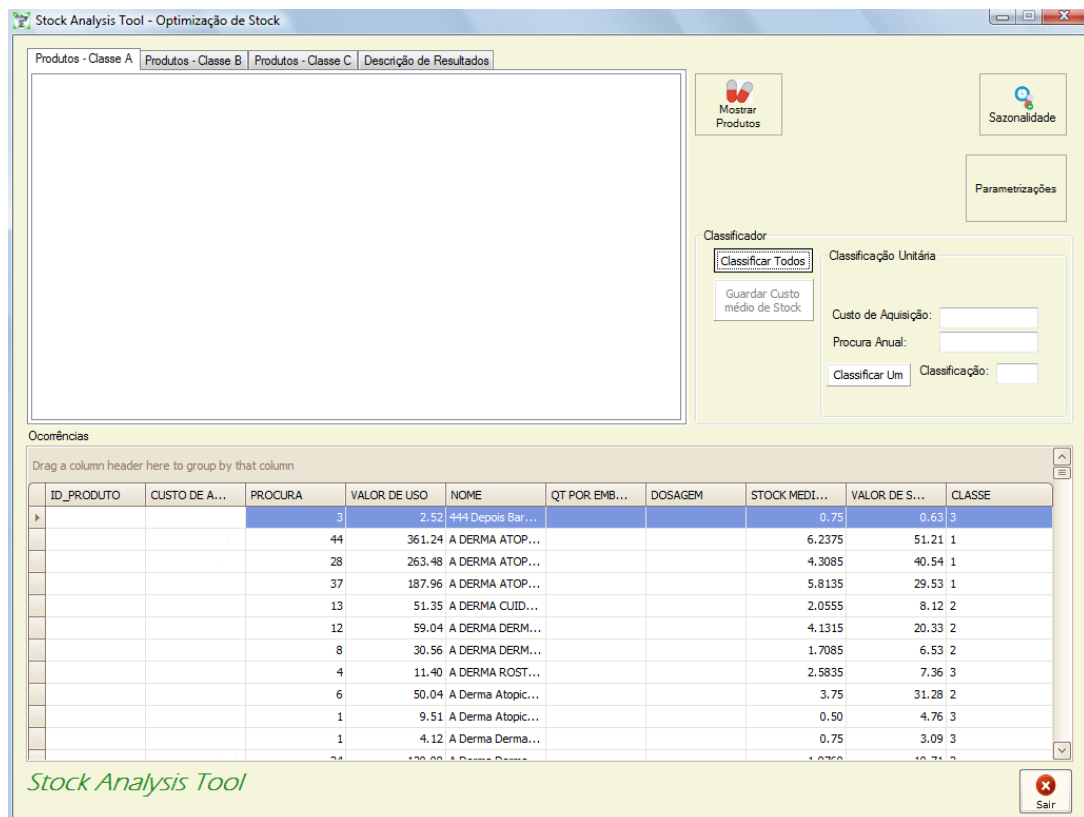
**Figura 3.6 – Stock Analysis Tool – Form Inicial**

De notar que os produtos são mostrados ao utilizador divididos em três separadores. Cada separador refere-se a uma única classe de produtos (A, B ou C).

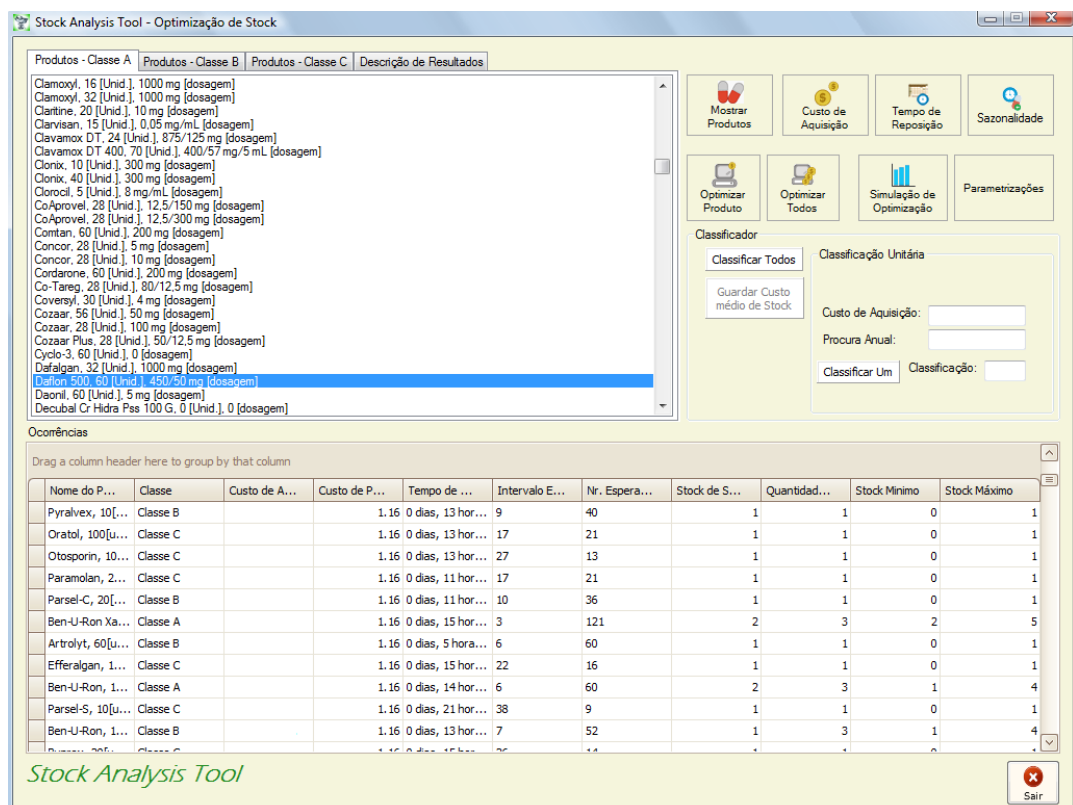
Na figura 3.7, mostra-se um exemplo dos resultados obtidos com a funcionalidade de classificação dos produtos. Tal como já foi dito anteriormente, devido a compromissos para com a farmácia que forneceu os dados alguns dados não podem ser mostrados, tais como código de produtos da base de dados e custos de aquisição, entre outros. Por esta razão, nos resultados mostrados não estão disponíveis o código do produto e o seu custo de aquisição.

Na figura 3.8 mostram-se os resultados relativos à *optimização* de todos os produtos em simultâneo. De notar que a *optimização* de um único produto é mostrada na mesma janela e de igual forma que a funcionalidade para todos os produtos.





**Figura 3.7 – Stock Analysis Tool – Classificação dos Produtos**



**Figura 3.8 – Stock Analysis Tool – Optimização de Todos os Produtos**

Finalmente, do *form* inicial, presente na figura 3.6, é ainda possível aceder a 3 funcionalidades adicionais:

- Simulação de *Optimização*, onde é efectuada a simulação da evolução dos *stocks* assim como os ganhos ou perdas obtidos com os parâmetros *optimizados* pela ferramenta SAT. Os *forms* associados a esta funcionalidade podem ser observados nas figuras 3.2 e 3.3, já anteriormente apresentados;
- Parametrizações, através das quais se efectua o cálculo do custo de armazenagem e do custo de processamento das encomendas. O *form* através do qual se efectuem estas parametrizações (já descritas na secção 3.3.2.2) pode ser observado na figura 3.9;

Stock Analysis Tool - Parametrizações

Armazém - Despesas com Limpeza  
horas/semana   
€/hora  €

Espaço Alocado - Renda  
Renda Mensal Total  €  
Área da Farmácia - Armazém [%]  %

Armazém de Produtos  
Salário do Funcionário  €  
Tempo despendido para armazenagem  minutos

Factura de Electricidade  
armazém [€/mês]  € ☒

Custo de Armazenagem:  <=>  % do valor de stock

Custo de Processamento de Encomenda:  €

**Figura 3.9** – *Stock Analysis Tool* - Parametrizações

- Sazonalidade dos Produtos, através da qual é efectuada a análise de sazonalidade dos produtos. Os *forms* relativos a esta funcionalidade podem ser observados nas figuras 3.4 e 3.5, na secção 3.4.2 deste documento.

# CAPÍTULO 4

## Conclusões e Trabalho Futuro

---

No final deste trabalho é possível dizer que o modelo Nível de Encomenda, quando comparado com os parâmetros de *stock* praticados pelo sistema em funções na farmácia (SF), melhora a gestão dos *stocks* do estabelecimento que o utilize, uma vez que se espera atingir, no caso da farmácia em estudo, ganhos aproximados de 1200€ ao fim do ano em que se vai utilizar os novos parâmetros de *stock*.

Relativamente ao classificador implementado e tal como é descrito na secção 3.2.2, dos 2636 produtos testados, as regras de classificação falharam na análise de apenas 54 desses produtos, ou seja, conclui-se que 2% dos produtos classificados são classificados incorrectamente. Este resultado é bastante baixo donde se conclui que o classificador implementado é uma boa solução para se ter uma noção da importância de cada produto para o negócio. Note-se ainda que este classificador, de acordo com a Análise ABC, deve ser efectuado no início de cada ano comercial pois nesse momento tem-se o histórico de todo o ano anterior obtendo-se assim as regras de forma mais adequada para o ano que está a iniciar.

De qualquer modo este trabalho não se encontra terminado, muito pelo contrário. Muitas variáveis não estavam acessíveis, factos relativos ao funcionamento da farmácia que contribuiriam de forma positiva para se alcançar melhores resultados e que se descrevem seguidamente.

Um dos factos importantes aos quais não se tinha acesso, ao longo deste trabalho, relaciona-se com o planeamento do horário semanal da farmácia. Ou seja, apenas se tem acesso a amostras dos movimentos não se sabendo em que dias a farmácia esteve aberta 24 horas por dia. Seria

assim importante ter acesso a este planeamento pois o cálculo de procura esperada seria efectuado com maior exactidão assim como a previsão de recepção de encomendas seria mais correcta. Ou seja, deste modo os resultados da optimização seriam mais fiáveis e deste modo os resultados da simulação também.

Outra falha nos dados fornecidos pelo SF prende-se com a falta de informação acerca da procura não satisfeita pela farmácia pois, mais uma vez, só se tem acesso aos movimentos registados. A procura não satisfeita acarreta prejuízos ao nível de insatisfação dos clientes que pode mesmo levar à perda desses clientes. Se se tivesse acesso a estes dados, certamente o sistema SF possuiria maior número de roturas de *stock* o que faria com que o ganho final do SAT fosse mais significativo. Seria assim importante, no futuro, ter um sistema de gestão da farmácia que registasse a procura não satisfeita.

Outro problema a melhorar prende-se com o facto de não se ter a noção de quando houve promoções da parte dos fornecedores e em que condições, ou seja, não se tem acesso ao preço real que se pagou por uma unidade de produto.

Com esta implementação, a *optimização* deve ser efectuada no início de cada ano comercial, de acordo com o próprio modelo Nível de Encomenda cuja procura média a calcular deve ser a procura diária ao longo de um ano. Uma das formas de melhorar a eficácia desta *optimização* é efectuá-la mais vezes ao longo do ano e de forma diferente, mais concretamente, no início de cada mês. Deste modo, supondo que se está no início do mês 3 do ano Y, então a *optimização* deveria ser feita com dados do mês anterior (2) do ano Y juntamente com os meses 3 e 4 do ano Y-1, pois assim a optimização seria apenas e só relativa ao período em estudo (neste caso seria relativamente a Fevereiro, Março e Abril), sendo deste modo os resultados mais fiáveis pois as variáveis das quais depende o modelo são também mais coerentes com o período corrente. Assim o modelo teria de sofrer adaptações

em relação ao período em análise, mais concretamente, relativamente à procura média diária que no novo período apresentado seria relativamente a 3 meses e não em relação a um ano.

O módulo implementado relativo à sazonalidade dos produtos foi efectuado na fase final desta tese, podendo ser claramente melhorado. A análise da sazonalidade, no que diz respeito à gestão de *stocks* de uma farmácia, é muito importante uma vez que, e por exemplo, em cada estação do ano existem doenças próprias assim como hábitos dos consumidores. No futuro, este módulo poderá ser melhorado permitindo que o modelo de *optimização* de *stocks* seja utilizado de forma diferente para os produtos sazonais, ou seja, no início do período de sazonalidade destes produtos, o modelo deve ser aplicado para esse período. Mal acabe o período de sazonalidade dos produtos o modelo deve ser novamente aplicado de forma a actualizar os parâmetros para este novo período no qual a procura dos produtos sazonais é escassa. Como se pode observar pelos resultados obtidos, na secção 3.4.2.1 e por exemplo, para a estação do Outono existem produtos, cujas vendas apenas ocorrem nesse período, ou seja, implementando esta ideia, seria possível ter poucos itens, nas restantes estações do ano, para os produtos sazonais, poupando-se assim espaço de armazenagem que ficaria livre para itens de outros produtos que estivessem mais sujeitos a roturas assim como em encomendas desses produtos.

No futuro seria importante a criação de um sistema de gestão de farmácias que considere as variáveis em falta apresentadas neste capítulo assim como as funcionalidades utilizadas na ferramenta SAT e seus melhoramentos aqui indicados, entre outras funcionalidades adicionais:

- Módulo com sistema de previsões de vendas (procura) que possibilita a existência de um sistema de alarmes para aconselhar o responsável pelo *stock* da farmácia a processar uma nova encomenda pois existe risco de rotura para o produto em causa;

- Como medida de marketing criar um módulo que dê indicações relativamente à distribuição dos produtos em *stock* pelo armazém e pelo espaço de loja (parteleiras/expositores, montra, etc.). Deste modo seriam necessárias as dimensões exactas destes espaços de alocação de *stock* e das embalagens dos produtos. A procura esperada dos produtos assim como se estes são sazonais para o período em que se efectua esta análise serão variáveis importantes a considerar;
- Estudo e implementação de políticas de descontos por parte da farmácia. Possibilidade de efectuar uma relação entre o *stock* e os prazos de validade de cada item;
- Análise dos melhores fornecedores. Conceber um módulo que de acordo com os tempos médios de reposição por parte dos fornecedores para cada produto ou de promoções dos mesmos, e no momento de efectuar uma encomenda, sugira ao responsável pelas encomendas qual o melhor fornecedor.

## Referências Bibliográficas

---

- [w3schoolsWS, 2010] Web Services Tutorial, <http://www.w3schools.com/webservices/default.asp>, acessado a 28 de Janeiro 2010.
- [msdnWS, 2010] Web Services by Microsoft MSDN, <http://msdn.microsoft.com/en-us/library/aa579187%28v=EXCHG.80%29.aspx>, acessado a 28 de Janeiro de 2010.
- [JavaTutorials, 2010] The Java Tutorials, Object-Oriented Programming Concepts, <http://download.oracle.com/javase/tutorial/java/concepts/index.html>, acessado a 28 de Janeiro de 2010.
- [Normal, 2010] Tabela II - Distribuição Normal Padrão, [ftp://www.puc-campinas.edu.br/pub/professores/ceatec/nreggiani/Mestrado%20-%20Gestao/Aula4\\_Estat%EDstica/TabelaNormal.pdf](ftp://www.puc-campinas.edu.br/pub/professores/ceatec/nreggiani/Mestrado%20-%20Gestao/Aula4_Estat%EDstica/TabelaNormal.pdf), Pontifícia Universidade Católica de Campinas, Brasil, acessado a 5 de Janeiro de 2010.
- [Gang, 2008] Gang, T., Kai, C., Bei, S. (2008). The Research and Application of Business Intelligence System in Retail Industry, Proceedings of the IEEE International Conference on Automation and Logistics, Qingdao, China, Setembro de 2008.
- [Santos, 2005] Santos, J. (2005). Programação Linear Inteira, Acetatos de Apoio. [http://www.estv.ipv.pt/PaginasPessoais/jsantos/InvOperGCP/AcetatosIO\\_GCP2005\\_2006.pdf](http://www.estv.ipv.pt/PaginasPessoais/jsantos/InvOperGCP/AcetatosIO_GCP2005_2006.pdf), acessado a 14 de Dezembro de 2010.
- [Heizer, 2004] Heizer, J., Render, B. (2004). Operations Management, Seventh Edition, International Edition, Pearson, Prentice Hall, pág 449-486.
- [Costa, 2002] Costa, R. (2002). Elementos de Apoio às Aulas de Introdução à Investigação Operacional, Licenciatura em Matemática, Departamento de Matemática, FCT/UNL.
- [Hartwig, 2002] Introduction To Web Services by Hartwig Gunzer, March 2002, [http://www.daimi.au.dk/~thomasr/Wearable/intro to web services wp.pdf](http://www.daimi.au.dk/~thomasr/Wearable/intro%20to%20web%20services%20wp.pdf), acessado a 28 de Janeiro de 2010.
- [Kimball, 2002] Kimball, R., Ross, M. (2002). The Data Warehouse Toolkit, Second Edition, The Complete Guide to Dimensional Modeling, Wiley, pág. 1-16.

- [Hillier, 2001] Hillier, F., Lieberman, G. (2001). Introduction To Operations Research, Seventh Edition, McGraw Hill, pág. 935-987.
- [Tanwari, 2000] Tanwari, A., Lakhari, A., Shaikh, G. (2000). ABC Analysis as an Inventory Control Technique, Quaid-E-Awam University Research Journal of Engineering, Science & Technology Volume 1, No. 1, Janeiro-Junho de 2000.
- [Carravilla, 2000] Gestão de Stocks, Acetatos de Apoio, da Faculdade de Engenharia da Universidade do Porto, <http://repositorio.up.pt/aberto/bitstream/10216/579/2/Gest%C3%A3o%20de%20stocks.pdf>, acedido a 21 de Novembro de 2009.
- [Carravilla, 1997] Gestão de Stocks, Elementos de Apoio, da Faculdade de Engenharia da Universidade do Porto, <http://repositorio.up.pt/aberto/bitstream/10216/569/2/Gest%C3%A3o%20de%20Stocks%20%20Texto%20de%20>, acedido a 21 de Novembro de 2009.
- [Tavares, 1996] Tavares L., Oliveira R., Themido I. e Correia F. (1996). Investigação Operacional, McGraw Hill, pág. 153-208.
- [Fonseca, 1994] Fonseca J. (1994). Indução de Árvores de Decisão – HistClass – Proposta de um Algoritmo Não Paramétrico, pág. 19-33.



# ANEXO A

## Breve Resumo de Especificações das Bases de Dados Utilizadas

---

Nesta secção, é feita uma breve descrição das bases de dados utilizadas para a concepção desta tese, mais concretamente, das respectivas tabelas criadas para este efeito assim como das respectivas funções ou procedimentos implementados.

### A.1. Tabelas Auxiliares

Previamente à instalação da aplicação é necessário assegurar que exista uma tabela auxiliar presente na base de dados fornecida pelo cliente. A tabela referida é então a seguinte:

#### **Tabela “AUXILIARDEMAND”**

Esta tabela é utilizada para guardar a informação referente à procura que se verificou para vários produtos. É utilizada no processo de optimização de todos os produtos em simultâneo, mais concretamente, para calcular o respectivo desvio padrão para os mesmos produtos, guardando então este último na tabela “AUXILIARSA” que a seguir se explica.

A tabela “AUXILIARDEMAND” é constituída pelos seguintes atributos:

- PRODID – Código do produto, que deverá ser do tipo NUMBER(17,0);
- VENDASDT – Data referente à venda verificada para o produto em questão, que deverá ser do tipo DATE;
- DEM – Quantidade transaccionada na data respectiva, que deverá ser do tipo NUMBER.

Na base de dados própria da aplicação *Stock Analysis Tool* há três tabelas, também estas importantes para o desenvolvimento do processo de *optimização* dos produtos em estudo. Estas três tabelas são as seguintes:

## Tabela “PRODUCT”

Esta tabela é utilizada para guardar a principal informação recolhida depois de se efectuar a optimização de um ou de todos os produtos, sendo constituída pelos seguintes atributos:

- ID – código do produto que é a chave primária desta tabela e que deve ser do tipo BIGINT;
- PRODUCT\_CLASS\_FK – chave estrangeira referente à tabela PRODUCT\_CLASS e que identifica a classe a que pertence o produto em causa. Este atributo deve ser do tipo BIGINT;
- ECO\_ORDER\_QUANTITY – quantidade económica a encomendar resultante do processo de *optimização*. Este atributo deve ser do tipo INT;
- MIN\_STOCK – valor do *stock* mínimo obtido após o processo de *optimização*. Este atributo deve ser do tipo INT;
- MAX\_STOCK – valor do *stock* máximo obtido após o processo de *optimização*. Este atributo deve ser do tipo INT;
- NOME – nome do produto, que deve ser do tipo NVARCHAR2(255);
- QT – número de comprimidos que constituem a embalagem, que deve ser do tipo INT;
- DOSE – dose química referente a cada comprimido da embalagem, que deve ser do tipo NVARCHAR(50);
- ORDER\_GAP – intervalo de dias esperado entre encomendas. Este atributo deve ser do tipo NVARCHAR(255);
- NR\_ORDERS\_YEAR – número de encomendas que se espera fazer ao longo do ano analisado, cujo tipo deve ser NVARCHAR(255);
- SAFETY\_STOCK – *stock* de segurança de cada produto. Dever ser do tipo INT;
- REPOSITION\_TIME – tempo médio de reposição do produto em segundos, que deve ser do tipo INT;

- REPOSITION\_TIME\_STR – descrição do tempo médio de reposição do produto para apresentação ao utilizador, que deve ser do tipo NVARCHAR(255);
- ACQUISITION\_COST – custo de aquisição do produto, que deve ser do tipo FLOAT;
- AVERAGE\_DEMAND – procura média diária do produto. Deve ser do tipo FLOAT;
- AVERAGE\_STOCK – valor médio de *stock* do produto, que deve ser do tipo FLOAT;
- STOCK\_VALUE – valor de *stock* do produto para a farmácia, que deve ser do tipo FLOAT;
- SIF\_STK\_OUT – Número de quebras verificadas no Sifarma para os produtos simulados com verificação de lucros usando a ferramenta concebida nesta tese (SAT) . Este atributo deve ser do tipo INT;
- SAT\_STK\_OUT - Número de quebras verificadas com a optimização da SAT para os produtos simulados com verificação de lucros usando a ferramenta concebida nesta tese. Este atributo deve ser do tipo INT;
- GAIN15PERC – Ganhos ou Lucros verificados, contabilizando 15% de perdas em relação a clientes não satisfeitos, para o respectivo produto. Este atributo deve ser do tipo FLOAT;
- GAIN30PERC – Ganhos ou Lucros verificados, contabilizando 30% de perdas em relação a clientes não satisfeitos, para o respectivo produto. Este atributo deve ser do tipo FLOAT;
- SIMULATION\_TIME\_SET – *string* que mostra o periodo de análise para o produto em causa a quando da simulação de *optimização* desse produto. Este atributo deve ser do tipo NVARCHAR(255).

## **Tabela “PRODUCT\_CLASS”**

Nesta tabela tem-se os diferentes tipos de classes atribuídas aos produtos guardados, tendo como atributos:

- ID – Código da classe que é a chave primária desta tabela e que deve ser do tipo BIGINT;
- DESCRIPTION – Descrição da classe que deve ser do tipo NVARCHAR(255).

## **Tabela “PHARMACY”**

Nesta tabela é possível ter guardada informação específica a cada farmácia utilizadora desta ferramenta. Para além da chave que a indentifica (ID – do tipo BIGINT), do nome (PHARMACY\_NAME do tipo VARCHAR(50)), da morada (ADDRESS do tipo VARCHAR(50)), do telefone (TELEPHONE do tipo VARCHAR(20)), fax (FAX do tipo VARCHAR(20)) e e-mail (EMAIL do tipo VARCHAR(50)) respectivos, a razão pela qual se criou esta tabela diz respeito aos seguintes atributos que merecem uma explicação mais detalhada:

- WAREHOUSE\_COSTS – Custo de armazenagem da farmácia em questão que deve ser do tipo FLOAT;
- PERCENTAGE\_OF\_STK\_VALUE – Este atributo consiste na percentagem do valor de *stock* da farmácia a que equivale o atributo WAREHOUSE\_COSTS. Este atributo deve ser do tipo FLOAT;
- SETUP\_COST – custo de processamento de cada encomenda. Este atributo deve ser do tipo FLOAT.

## A.2. Funções ou Procedimentos PL/SQL Auxiliares

São também necessárias algumas funções pl/sql a ser adicionadas à base de dados fornecida pelo cliente.

Seguidamente, descreve-se sucintamente cada um(a) destes(as) procedimetos ou funções.

Comece-se pelas funções necessárias, que são as seguintes:

### ***SA\_CORRECT\_ACQ\_COST***

Esta função devolve o custo de aquisição correcto de um determinado produto. Caso o produto seja uma especialidade farmacêutica é retirado o valor do IVA, aplicado no produto, ao custo de aquisição que se encontra guardado na base de dados do cliente.

Esta função tem três argumento de entrada, *grg\_type* (diz se é especialidade farmacêutica ou não), *iva* (valor do iva aplicado no produto) e *preco* (custo de aquisição do produto presente na base de dados original) e a variável de saída é do tipo NUMBER.

### ***SA\_STD\_DEV***

Esta função devolve o desvio padrão da procura de um produto. Esta função tem três argumentos de entrada, *produto* (código do produto), *average* (procura média diária do produto) e *nrDays* (número de dias em que a farmácia esteve aberta) e a variável de saída é do tipo NUMBER.

### ***SA\_CONVERT\_REPTIME2STRING***

Esta função converte o tempo médio de reposição (fracção de um dia, por exemplo, 0,5) numa cadeia de caracteres com a descrição do mesmo (por exemplo, “0 dias, 12 horas e 0 minutos”).

Esta função tem como argumento de entrada, *rep\_time* (tempo médio de reposição em fracção de um dia) e a variável de saída é do tipo VARCHAR2.

### ***SA\_CONVERT\_SECS2HOURS***

Esta função converte um valor numérico, cuja grandeza é segundos, numa cadeia de caracteres que mostra esses segundos em termos de horas e minutos. Por exemplo, recebendo um valor de 5400 segundos é devolvida a seguinte cadeia de caracteres, “01:30:18”.

Esta função tem como argumento de entrada, *time\_set* (valor numérico que corresponde a um valor em segundos) e a variável de saída é do tipo VARCHAR2.

### ***SA\_PRODUCT\_AVG\_STOCK***

Esta função calcula o valor médio de *stock* de um determinado produto (*produto* – argumento de entrada do tipo NUMBER) num determinado ano (*yearAnalysis* – argumento de entrada do tipo VARCHAR2. Exemplo: “2006”). Deste modo, a variável de saída (valor médio de *stock*) é do tipo NUMBER.

Esta função é utilizada, por exemplo, no método que executa a classificação dos produtos tidos em *stock*.

Os **Procedimentos** necessários são descritos a seguir.

### ***SA\_STD\_DEV\_PROC***

Recorrendo às tabelas auxiliares AUXILIARSA e AUXILIARDEMAND calcula-se, para todos os produtos aí guardados, o desvio padrão da procura dos produtos em causa.

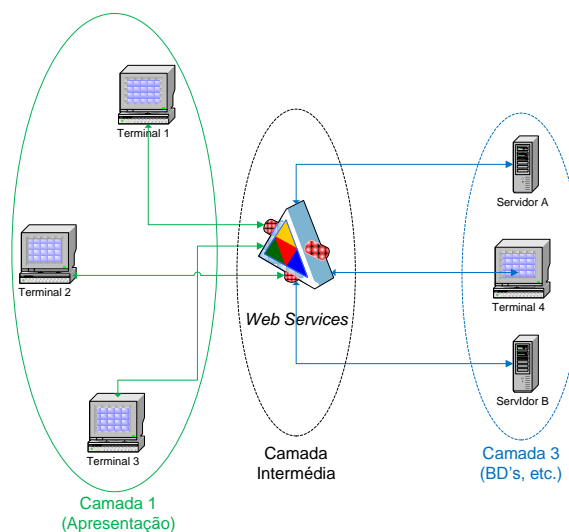
# ANEXO B

## Breve Descrição do *Web Service* Implementado

Nesta secção, é feita uma breve descrição do *Web Service* implementado para concepção desta tese, sendo inicialmente dada uma breve explicação acerca do conceito de *Web Services*.

### B.1. *Web Services* – Conceito

A partilha de informação é uma necessidade entre diferentes parceiros comerciais, as empresas e os seus clientes e até entre os departamentos de uma empresa. Esta partilha de informação é possível através dos *Web Services* que a disponibilizam de uma forma distribuída. Um sistema distribuído é possível dividindo a aplicação em duas partes: cliente e servidor. O servidor fornece os serviços e informação ao cliente que os consome. Este tipo de estrutura apresenta problemas ao nível de escalabilidade e por esta razão houve a necessidade de criar uma camada intermédia, presente na figura B.1. A 1ª camada é a camada de apresentação, a 2ª camada, a intermédia, contém a lógica e a 3ª camada gere os dados a utilizar.



**Figura B.1** – Exemplo de Arquitectura Distribuída

Os *Web Services* ([w3schoolsWS, 2010], [msdnWS, 2010]) são uma tecnologia que pode desempenhar as funções da camada intermédia, podendo ser publicados, localizados e invocados por intermédio da Internet.

Sempre que uma ou várias aplicações querem aceder a dados contidos nos servidores ou a funcionalidades de outras aplicações localizadas em outros terminais ligados à Internet e aos *Web Services*, é feito um pedido para satisfazer o serviço pedido que os *Web Services* disponibilizam contactando com os servidores/aplicações que guardam/desempenham tais dados/funções.

Mais concretamente, os *Web Services* são componentes de aplicação *Web* que comunicam através de protocolos abertos. A plataforma base é constituída pelas linguagens HTML (*Hyper Text Markup Language*) ou XML (*eXtensible Markup Language*) e pelo protocolo HTTP (*Hyper Text Transfer Protocol*).

O XML disponibiliza uma linguagem que pode ser usada tanto entre diferentes plataformas como entre diferentes linguagens e entre estes dois tipos em conjunto, sendo sempre perceptível entre cada um dos lados da comunicação. Permite ainda a codificação e decodificação dos dados trocados entre as aplicações ou plataformas, sendo o transporte destes dados assegurado pelo protocolo SOAP (*Simple Object Access Protocol*).

O protocolo HTTP é o protocolo de Internet mais usado.

Para além do protocolo SOAP, os *Web Services* são ainda constituídos por outros dois elementos, UDDI (*Universal Description Discovery and Integration*) e WSDL (*Web Services Description Language*). Seguidamente é feita uma breve descrição destes três elementos.

O SOAP é um protocolo baseado em XML que permite a troca de informação por HTTP, ou seja, torna possível o acesso de diferentes aplicações ou plataformas a *Web Services* disponíveis na rede global. Este protocolo tem como vantagens ser uma linguagem ou plataforma independente, simples e que permite a escalabilidade dos sistemas.



A WSDL, tal como o próprio nome indica, é uma linguagem baseada em XML que permite a localização e descrição de *Web Services*. Esta linguagem encontra-se normalizada pelo W3C.

O UDDI é um serviço onde as empresas podem procurar *Web Services* e guardar informação acerca dos mesmos. É ainda uma pasta de interfaces de *Web Services* que é descrita por WSDL e que comunica via SOAP. Adicionalmente, já se encontra implementada na plataforma *Microsoft .NET*.

Deste modo, optou-se pela utilização dos *Web Services* por razões de segurança porque consegue-se separar o nível de apresentação de uma aplicação (cliente) dos dados (servidor). Deveu-se ainda a razões de escalabilidade pois, deste modo, é possível adicionar diferentes aplicações ao sistema implementado desde que estas pretendam utilizar os serviços implementados.

## **B.2. Descrição dos *WebMethods* Implementados**

A descrição do *Web Service* e respectivos métodos (*WebMethods*) é efectuada nesta secção. Os primeiros três *WebMethods* referem-se a recolha directa de produtos e respectiva informação guardada em tabelas existentes nas bases de dados utilizadas ao longo deste trabalho.

### ***GetProductsClassifier***

Recorrendo à base de dados fornecida extrai-se todos os produtos referentes ao ano a analisar. Mais concretamente, a informação extraída do produto neste *WebMethod* são o código de produto, o custo de aquisição, a procura que o produto teve ao longo do ano, o valor de uso do produto para a farmácia, o nome do produto, a quantidade tida na embalagem, a dose química em cada unidade, o *stock* médio ao longo do ano em causa e respectivo valor de *stock*. Nesta extracção só são considerados produtos que tiveram procura maior ou igual a zero e cujo custo de aquisição seja positivo não nulo.

### ***GetProductsSA***

Após os produtos se encontrarem classificados, com este *WebMethod* extrai-se os produtos e respectivos dados de uma determinada classe. A informação extraída são o código do produto, o nome, a quantidade tida na embalagem, a dosagem e a classe a que o produto pertence.

### ***GetAllProductsSA***

Com este *WebMethod* extrai-se os produtos, mais concretamente, o seu código, o nome, a quantidade tida na embalagem e a dosagem, guardados na tabela Product. Este *WebMethod* é utilizado na funcionalidade da sazonalidade dos produtos.

Os *WebMethods* seguintes dizem respeito a acções sobre um único produto. São utilizados quer na *optimização* de um ou de todos os produtos em simultâneo.

### ***GetAvgRepTime***

Cálculo do tempo médio de reposição para um produto, sendo mostrado o seu valor quer na versão de fracção de dia quer na sua versão em forma de cadeia de caracteres com a descrição do número de dias, horas e minutos.

### ***GetAcqCost***

O propósito principal deste *WebMethod* é extrair o custo de aquisição de um único produto ao longo do ano analisado. Para tal efeito é também necessário extrair duas grandezas, o tipo de produto (se é especialidade farmacêutica ou não) e o IVA aplicado ao produto.

De notar ainda que para este efeito é utilizada a função em pl/sql, SA\_CORRECT\_ACQ\_COST, para retornar o correcto custo de aquisição. Para além do código do produto é ainda necessário se saber o ano a que diz respeito esta extracção.

***GetAvgDemand***

Extracção da procura média diária ao longo de um ano para um produto. É necessário saber o código do produto, saber o ano de análise e o número total de dias desse mesmo ano.

***TruncateDemand***

Os dados da tabela auxiliar AUXILIARDEMAND são apagados de forma a se guardar nova recolha de procura do produto para posterior cálculo do desvio padrão da procura.

***Demand***

Extraí-se a procura ao longo de um ano para um produto específico, guardando-se essa procura por dia em que se verificou movimento, na tabela AUXILIARDEMAND.

***GetStdDeviation***

Extracção do desvio padrão da procura média ao longo de um ano para um produto. Além do código do produto é necessário saber a média referida e o número total de dias do ano.

***GetPharmacySetupCost***

*Web Method* através do qual se extrai o custo de processamento de qualquer encomenda efectuada.

***ReadOptData2***

Neste *WebMethod* extrai-se informação relativa à *optimização* ou análise dos produtos. A informação extraída são o nome do produto, a classe a que pertence, o custo de aquisição, o custo de processamento das encomendas, o tempo de reposição, o intervalo entre encomendas, número de encomendas esperadas ao longo do ano, o *stock* de segurança, a quantidade económica a encomendar e os *stocks* mínimo e máximo. Este *Web Method* é apenas utilizado na funcionalidade de *optimização* de todos os produtos em simultâneo.

Após se ter a informação guardada na base de dados do SAT, ou seja, na base de dados (*SQL Server*) onde é guardada a informação da *optimização*, é então necessário fazer-se algumas inspecções a essa mesma tabela. Os *WebMethods* seguintes referem-se a leituras, inserções ou alterações à tabela de produtos dessa mesma base de dados.

### ***ReadRepositionTimeFromDb***

Neste *WebMethod* é efectuada a leitura do tempo médio de reposição de um produto. Este *WebMethod* é utilizado quando se pretende efectuar a simulação do comportamento dos *stocks* considerando os valores de *optimização* devolvidos pelo SAT.

### ***ReadStkMinMaxFromDb***

Este *WebMethod* é também utilizado na simulação do comportamento dos *stocks*, extraindo os valores dos parâmetros de *optimização* (*stock* mínimo e máximo) para um produto.

### ***InsertProductsInDB***

Após a classificação dos produtos, estes são inseridos na tabela *PRODUCT* da base de dados da SAT (*SQL Server*). Nota: A informação prévia mantida nesta tabela é apagada. Tem-se assim que ter, previamente, o conjunto de dados a guardar que são dados através um objecto *datatable*, o custo de processamento das encomendas e o número total de dias do ano em questão.

### ***UpdateProductStk***

Neste *WebMethod* actualiza-se a tabela *PRODUCT*, mais concretamente os atributos relativos aos *stocks* mínimo e máximo, tempo de reposição e custo de aquisição. É o método utilizado no final do processo de *optimização* de cada produto.

### ***UpdateProductRepTime***

Este *WebMethod* actualiza o atributo *REPOSITION\_TIME* da tabela *PRODUCT*, ou seja, actualiza o tempo de reposição de um produto.

### ***UpdateProductsSimulParam***

Este *WebMethod*, actualiza os parâmetros relativos à simulação do *stock* de todos os produtos em simultâneo, ou seja, as roturas verificadas nos dois sistemas SIF e SAT, os ganhos obtidos considerando 15% e 30% de perda de clientes não satisfeitos, respectivamente, e finalmente o período de tempo analisado.

Após os processos de classificação e optimização, a que se referem directamente alguns dos métodos apresentados anteriormente, procedeu-se à simulação e comparação dos comportamento de evolução dos *stocks* considerando os *stocks* mínimo e máximo iniciais e os valores após a *optimização*. Para tal é necessário recorrer aos *WebMethods* apresentados a seguir.

### ***GetStkMonthSimSIF***

Extrai-se com este *WebMethod* os dados necessários para efectuar a simulação da evolução dos *stocks* do sistema implementado na ferramenta de gestão do cliente, ou seja, considerando os mínimos e máximos originais. Para além do produto a analisar, é necessário saber o intervalo de tempo de análise.

### ***GetStkMonthSimSAT***

Extrai-se com este *WebMethod* os dados necessários para efectuar a simulação da evolução dos *stocks* do sistema implementado no SAT, ou seja, considerando os mínimos e máximos *optimizados*. Para além do produto a analisar, é necessário saber a data em causa.

### ***ReadAvgDemandFromDB***

Para um determinado produto é extraído o valor da procura média tida em base de dados. Este método é utilizado na simulação dos *stocks*.

***ReadInitialStockForAnalysis***

São obtidas todas as ocorrências de alterações de *stock* antes da data de início do período de análise, ordenando estas ocorrências de forma decrescente relativa à data de movimento.

***ReadStkMinMaxSif***

Para um determinado produto são extraídos os seus *stocks* mínimo e máximo. Este método é utilizado na simulação dos *stocks*.

***GetProductSaleValue***

Com este *WebMethod* e mediante a especificação do produto e do intervalo de análise, é processado e devolvido o preço de venda do produto analisado.

***GetPercentageOfStkValue***

Este *WebMethod* extrai, para uma determinada farmácia, a percentagem do seu valor total de *stock* a que corresponde o seu custo de armazenagem.

***GetGainDistributionByClass***

Este *WebMethod* extrai a distribuição dos ganhos obtidos pelas três classe consideradas, A, B e C.

***GetTopTenProductsByGain***

Este *WebMethod* extrai os dez produtos para os quais se consegue maior ganho.

***GetPrimaryDataFromLastSimulation***

Este *WebMethod* extrai a informação a mostrar ao utilizador relativamente à simulação dos *stocks* efectuada.

***GetTotalGains15and30PercLosses***

Este *WebMethod* extrai os ganhos e perdas obtidos na ultima simulação efectuada.

Na funcionalidade em que se efectuam parametrizações de dados relativos à farmácia em estudo, ou seja, em que se calcula o custo de armazenagem e a percentagem do valor total de *stock* a que corresponde, para além do cálculo do custo de processamento das encomendas, são utilizados dois *WebMethods* que se descrevem a seguir.

### ***GetPharmacyStockValueSA***

*WebMethod* necessário para o cálculo da percentagem de valor de *stock* a que corresponde o custo de armazenagem da farmácia. Com este *WebMethod* efectua-se então a extracção do valor total de *stock* da farmácia.

### ***UpdatePharmacyStorageSetupCost***

*WebMethod* necessário para guardar os valores dos três parâmetros calculados na funcionalidade em questão.

Relativamente à análise da sazonalidade dos produtos são usados os seguintes *WebMethods* para além do *WebMethod* GetAllProductsSA já descrito anteriormente.

### ***GetSeasonalityResultsByMonth***

São extraídos os produtos que são considerados sazonais no período escolhido, considerando os meses escolhidos.

### ***GetSeasonalityResultsBySeason***

São extraídos os produtos que são considerados sazonais no período escolhido, ou seja, numa das quatro estações do ano disponíveis na aplicação.

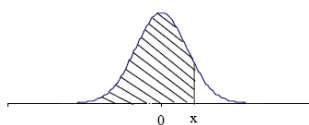
### ***GetYearSalesByProduct***

Para cada produto é extraído o volume total de vendas ao longo do ano em causa.

# ANEXO C

## Distribuição Normal Reduzida

Pela tabela C.1 da distribuição Normal Reduzida dos valores de probabilidade,  $P(Z \leq x)$  tira-se o valor de  $x$  que corresponde ao  $Z$  [Normal, 2010]. Como exemplo considere-se  $ns = 97\%$ .



**Figura C.1** – Exemplo de Distribuição Normal,  $P(Z \leq x)$

x	0	0,01	0,02	0,03	0,04	0,05	0,06	0,07	0,08	0,09
0	0,5000	0,5040	0,5080	0,5120	0,5160	0,5199	0,5239	0,5279	0,5319	0,5359
0,1	0,5398	0,5438	0,5478	0,5517	0,5557	0,5596	0,5636	0,5675	0,5714	0,5753
0,2	0,5793	0,5832	0,5871	0,5910	0,5948	0,5987	0,6026	0,6064	0,6103	0,6141
0,3	0,6179	0,6217	0,6255	0,6293	0,6331	0,6368	0,6406	0,6443	0,6480	0,6517
0,4	0,6554	0,6591	0,6628	0,6664	0,6700	0,6736	0,6772	0,6808	0,6844	0,6879
0,5	0,6915	0,6950	0,6985	0,7019	0,7054	0,7088	0,7123	0,7157	0,7190	0,7224
0,6	0,7257	0,7291	0,7324	0,7357	0,7389	0,7422	0,7454	0,7486	0,7517	0,7549
0,7	0,7580	0,7611	0,7642	0,7673	0,7704	0,7734	0,7764	0,7794	0,7823	0,7852
0,8	0,7881	0,7910	0,7939	0,7967	0,7995	0,8023	0,8051	0,8078	0,8106	0,8133
0,9	0,8159	0,8186	0,8212	0,8238	0,8264	0,8289	0,8315	0,8340	0,8365	0,8389
1	0,8413	0,8438	0,8461	0,8485	0,8508	0,8531	0,8554	0,8577	0,8599	0,8621
1,1	0,8643	0,8665	0,8686	0,8708	0,8729	0,8749	0,8770	0,8790	0,8810	0,8830
1,2	0,8849	0,8869	0,8888	0,8907	0,8925	0,8944	0,8962	0,8980	0,8997	0,9015
1,3	0,9032	0,9049	0,9066	0,9082	0,9099	0,9115	0,9131	0,9147	0,9162	0,9177
1,4	0,9192	0,9207	0,9222	0,9236	0,9251	0,9265	0,9279	0,9292	0,9306	0,9319
1,5	0,9332	0,9345	0,9357	0,9370	0,9382	0,9394	0,9406	0,9418	0,9429	0,9441
1,6	0,9452	0,9463	0,9474	0,9484	0,9495	0,9505	0,9515	0,9525	0,9535	0,9545
1,7	0,9554	0,9564	0,9573	0,9582	0,9591	0,9599	0,9608	0,9616	0,9625	0,9633
1,8	0,9641	0,9649	0,9656	0,9664	0,9671	0,9678	0,9686	0,9693	0,9699	0,9706
1,9	0,9713	0,9719	0,9726	0,9732	0,9738	0,9744	0,9750	0,9756	0,9761	0,9767
2	0,9772	0,9778	0,9783	0,9788	0,9793	0,9798	0,9803	0,9808	0,9812	0,9817
2,1	0,9821	0,9826	0,9830	0,9834	0,9838	0,9842	0,9846	0,9850	0,9854	0,9857
2,2	0,9861	0,9864	0,9868	0,9871	0,9875	0,9878	0,9881	0,9884	0,9887	0,9890
2,3	0,9893	0,9896	0,9898	0,9901	0,9904	0,9906	0,9909	0,9911	0,9913	0,9916
2,4	0,9918	0,9920	0,9922	0,9925	0,9927	0,9929	0,9931	0,9932	0,9934	0,9936
2,5	0,9938	0,9940	0,9941	0,9943	0,9945	0,9946	0,9948	0,9949	0,9951	0,9952
2,6	0,9953	0,9955	0,9956	0,9957	0,9959	0,9960	0,9961	0,9962	0,9963	0,9964
2,7	0,9965	0,9966	0,9967	0,9968	0,9969	0,9970	0,9971	0,9972	0,9973	0,9974
2,8	0,9974	0,9975	0,9976	0,9977	0,9977	0,9978	0,9979	0,9979	0,9980	0,9981
2,9	0,9981	0,9982	0,9982	0,9983	0,9984	0,9984	0,9985	0,9985	0,9986	0,9986
3	0,9987	0,9987	0,9987	0,9988	0,9988	0,9989	0,9989	0,9989	0,9990	0,9990
3,1	0,9990	0,9991	0,9991	0,9991	0,9992	0,9992	0,9992	0,9992	0,9993	0,9993
3,2	0,9993	0,9993	0,9994	0,9994	0,9994	0,9994	0,9994	0,9995	0,9995	0,9995
3,3	0,9995	0,9995	0,9995	0,9996	0,9996	0,9996	0,9996	0,9996	0,9996	0,9997
3,4	0,9997	0,9997	0,9997	0,9997	0,9997	0,9997	0,9997	0,9997	0,9997	0,9998

x	1.282	1.645	1.96	2.326	2.576	3.090	3.291	4.891	4.417
F(x)	0.90	0.95	0.975	0.99	0.995	0.999	0.9995	0.99995	0.999995
2[1-F(x)]	0.20	0.10	0.05	0.02	0.01	0.002	0.001	0.0001	0.00001

**Tabela C.1** – Tabela da Distribuição Normal Reduzida

Assim da tabela C.1 conclui-se que para um  $ns = 97\%$  tem-se  $Z = 1,89$ .



# ANEXO D

## Modelos Estocásticos – Exemplos

---

Neste anexo são explicitados dois exemplos demonstrativos da aplicabilidade dos modelos de *optimização* estocásticos analisados na secção 2.2.2, ou seja, Modelo por Revisão Cíclica e Modelo de Nível de Encomenda.

Os exemplos respectivos aos dois modelos referem-se ao seguinte enunciado de problema que se apresenta a seguir.

### Enunciado do Problema

*Considere-se um armazém que distribui uma marca de comida para gatos. Relativamente a esse produto são conhecidos os seguintes dados:*

- procura média diária = 225 pacotes
- tempo de reaprovisionamento = 3 dias
- desvio padrão da procura diária = 175 pacotes
- nível de serviço pretendido = 95%
- custo de encomenda = 18 u.m. por encomenda
- custo por pacote (unitário) = 8 u.m.
- taxa de armazenamento =  $0.2 \times$  custo de aquisição

*Considere-se ainda que o armazém está aberto 250 dias por ano. Efectuar a optimização de stock para o período de um ano.*

## D.1. Revisão Cíclica

Comece-se assim por determinar o intervalo de tempo entre encomendas. Uma vez que se pretende efectuar a optimização para o período de um ano então há que considerar a procura anual do produto,  $d_{\text{anual}}$ :

$$T = \frac{Q^*}{d_{\text{anual}}} = \sqrt{\frac{2A}{d_{\text{anual}} \times c_2}} = \sqrt{\frac{2 \times 18}{250 \times 225 \times 0.2 \times 8}} = 0.02$$

O que em termos diários corresponde a  $0.02 \times 250 = 5$  dias.

Obtêm-se então os seguintes dados:

$$d \times (L + T) = 225 \times (3 + 5) = 1800 \text{ pacotes}$$

$$\sigma' = \sqrt{(3 + 5)} \times 175 = 495 \text{ pacotes}$$

$$ns = 95\% \Rightarrow z = 1.65 \text{ (ver Anexo C)}$$

$$s = z \times \sigma' = 1.65 \times 495 = 817 \text{ pacotes}$$

$$M = d \times (L + T) + z \times \sigma' = 2617$$

Dos dados anteriores tiram-se assim as seguintes conclusões quanto às medidas de *optimização* a considerar:

- Rever o *stock* de 5 em 5 dias (dias operacionais) e encomendar por forma a se ficar em *stock* com 2617 pacotes.
- *Stock* de segurança será de 817 pacotes.
- Assim por ano serão colocadas  $\frac{250 \text{ dias}}{5 \text{ dias}} = 50$  encomendas por ano.

## D.2. Nível de Encomenda

Uma vez que se pretende efectuar a optimização para o período de um ano então há que considerar a procura anual do produto,  $d_{\text{anual}}$ . Obtêm-se assim os seguintes dados:

$$Q^* = \sqrt{\frac{2 A d_{\text{anual}}}{c_2}} = \sqrt{\frac{2 \times 18 \times 250 \times 225}{0.2 \times 8}} = 1125 \text{ pacotes por encomenda.}$$

O intervalo de encomendas virá:

$$T = \frac{Q^*}{d_{\text{anual}}} = \sqrt{\frac{2 A}{d_{\text{anual}} \times c_2}} = \sqrt{\frac{2 \times 18}{250 \times 225 \times 0.2 \times 8}} = 0.02$$

O que em termos diários corresponde a  $0.02 \times 250 = 5$  dias.

Obtêm-se então os seguintes dados:

$$d \times L = 225 \times 3 = 675 \text{ pacotes}$$

$$\sigma = \sqrt{3} \times 175 = 303 \text{ pacotes}$$

$$ns = 95\% \Rightarrow z = 1.65$$

$$s = z \times \sigma \cong 500 \text{ pacotes}$$

$$E = d \times L + z \times \sigma = 1175$$

Dos dados anteriores tiram-se assim as seguintes conclusões quanto às medidas de *optimização* a considerar:

- Colocar uma encomenda de 1125 pacotes sempre que o *stock* passe abaixo de 1175 pacotes.
- *Stock* de Segurança será de, aproximadamente, 500 pacotes.

Assim por ano serão colocadas  $\frac{250 \text{ dias}}{5 \text{ dias}} = 50$  encomendas por ano.

# ANEXO E

## Modelo Nível de Encomenda – Método Iterativo

---

Tal como no modelo Nível de encomenda apresentado na secção 2.2.2.2 o custo de encomenda representa-se por  $A$  e o custo de aquisição é dado por  $C_1Q$  sendo  $C_1$  o custo unitário de aquisição do produto.

### E.1. Optimização Global: Procura Aleatória Contínua e Vendas Diferidas

Sendo o nível de existências dado por  $\frac{1}{2}(Q - S) + \frac{1}{2}S = \frac{Q}{2} + S = \frac{Q}{2} + E - \mu$  e representando  $C_2$  o custo de manter em stock uma unidade de artigo durante uma unidade de tempo então o custo de posse por ciclo é dado pela equação E.1.

$$\text{Custo de posse} = C_2 \left( \frac{Q}{2} + E - \mu \right) \bar{T} \quad (\text{E. 1})$$

Supondo que o custo de vendas diferidas é proporcional à quantidade em falta e se  $C_3$  representa o custo por unidade de produto em falta então o custo de vendas diferidas por ciclo é dado pela equação E.2.

$$\text{Custo de vendas diferidas} = C_3 \int_E^{+\infty} (x - E)h(x) dx \quad (\text{E. 2})$$

Tem-se então que a quantidade óptima ( $Q^*$ ) a encomendar que permite a minimização do custo total por unidade de tempo é dada pela equação E.3 e o ponto de encomenda óptimo ( $E^*$ ) é dado pela equação E.4.

$$Q^* = \sqrt{\frac{2d(A + C_3 \int_{E^*}^{+\infty} (x - E^*)h(x) dx)}{C_2}} \quad (\text{E. 3})$$

$$\int_{E^*}^{+\infty} h(x) dx = \frac{C_2 Q^*}{C_3 d} \quad (\text{E. 4})$$

Com estas duas equações é possível então calcular os valores finais de  $Q^*$  e  $E^*$  através do processo iterativo:

1. Iniciar o processo com o cálculo de um valor de  $Q$  dado pela equação 2.57, presente na secção 2.2.2.2;
2. Utilizando este valor de  $Q$  calculado e a equação E.4 calcula-se então o respectivo valor de  $E^*$ ;
3. Determinar um novo valor de  $Q$  substituindo o valor de  $E$  (calculado em 2.) na equação E.3;
4. Voltar a 2.

Este processo pára assim que os novos valores óptimos encontrados sejam muito semelhantes aos encontrados na iteração anterior.

## E.2. Optimização Global: Procura Aleatória Contínua e Vendas Perdidas

Sendo o nível de existências dado por  $\frac{Q}{2} + E - \mu + \int_E^{+\infty} (x - E)h(x) dx$  e representando  $C_2$  o custo de manter em stock uma unidade de artigo durante uma unidade de tempo então o custo de posse por ciclo é dado pela equação E.5.

$$\text{Custo de posse} = C_2 \left( \frac{Q}{2} + E - \mu + \int_E^{+\infty} (x - E)h(x) dx \right) \bar{T} \quad (\text{E. 5})$$

Supondo que o custo de vendas perdidas é proporcional à quantidade em falta e se  $C_3$  representa o custo por unidade de produto em falta então o custo de vendas perdidas por ciclo é dado pela equação E.6.

$$\text{Custo de vendas perdidas} = C_3 \int_E^{+\infty} (x - E)h(x) dx \quad (\text{E. 6})$$

Tem-se então que a quantidade óptima ( $Q^*$ ) a encomendar que permite a minimização do custo total por unidade de tempo é dada pela equação E.7 e o ponto de encomenda óptimo é dado pela equação E.8.

$$Q^* = \sqrt{\frac{2 d (A + C_3 \int_{E^*}^{+\infty} (x - E^*) h(x) dx)}{C_2}} \quad (E. 7)$$

$$\int_{E^*}^{+\infty} h(x) dx = \frac{C_2 Q^*}{C_2 Q^* + C_3 d} \quad (E. 8)$$

Com estas duas equações é possível então calcular os valores finais de  $Q^*$  e  $E^*$  através do processo iterativo:

1. Iniciar o processo com o cálculo de um valor de  $Q$  dado pela equação 2.57, presente na secção 2.2.2.2;
2. Utilizando este valor de  $Q$  calculado e a equação E.8 calcula-se então o respectivo valor de  $E^*$ ;
3. Determinar um novo valor de  $Q$  substituindo o valor de  $E$  (calculado em 2.) na equação E.7;
4. Voltar a 2.

Este processo pára assim que os novos valores óptimos encontrados sejam muito semelhantes aos encontrados na iteração anterior.

# ANEXO F

## Análise ABC – Exemplo

---

Neste exemplo, pretende-se efectuar uma categorização ABC da lista de produtos, presentes na tabela E.1, para um período de tempo de um ano [Carravilla, 2000]. Para tal, reúne-se a informação necessária inicialmente, ou seja, o custo unitário de aquisição de cada um dos produtos da empresa assim como a procura anual dos mesmos. Calcula-se então o respectivo valor de uso de cada produto utilizando a equação 2.59, referida anteriormente na secção 2.3.

Codigo do artigo	Taxa de Procura Anual	Custo/unidade (milhares \$)	Valor de uso
P1	10000	13	130000
P2	5000	5	25000
P3	2000	6	12000
P4	1000	18	18000
P5	100	55	5500
P6	5000	8	40000
P7	2000	2	4000
P8	200	12	2400
P9	25000	10	250000

**Tabela F.1** – Exemplo de Análise ABC: Valor de Uso

Após o valor de uso de cada produto estar calculado efectua-se a ordenação dos produtos por ordem decrescente do seu valor de uso, tal como se mostra na tabela E.2.

Codigo do artigo	Taxa de Procura Anual	Custo/unidade (milhares \$)	Valor de uso
P9	25000	10	250000
P1	10000	13	130000
P6	5000	8	40000
P2	5000	5	25000
P4	1000	18	18000
P3	2000	6	12000
P5	100	55	5500
P7	2000	2	4000
P8	200	12	2400

**Tabela F.2** – Exemplo de Análise ABC: Ordenação Decrescente por Valor de Uso

Numera-se então, de  $1$  até  $n$ , os produtos ordenados anteriormente e calcula-se a percentagem do número de ordem no número total de artigos (coluna *% Total*).

Tome-se como exemplo o produto P6 que corresponde a uma percentagem de 33,33% do total de artigos, ou seja:

$$[\% \text{ total de artigos}]_{P6} = \frac{3}{9} \times 100\% \cong 33,33\%$$

Calcula-se ainda o acumulado e percentagem do total acumulado. Neste caso, o produto P6 corresponderá a um acumulado de 420000 (acumulado\_P1 + acumulado\_P2 + acumulado\_P3) e a uma percentagem de acumulado de 86,26%, ou seja:

$$[\% \text{ de acumulado}]_{P6} = \frac{\text{acumulado}_{P6}}{\text{acumulado}_{\text{total}}} \times 100\% = \frac{420000}{486900} \times 100\% \cong 86,26\%$$

Codigo do artigo	Nº de Ordem	% Total	Valor de uso	Acumulado	% Total	Classe ABC
P9	1	11.11	250000	250000	51.345	A
P1	2	22.22	130000	380000	78.045	A
P6	3	33.33	40000	420000	86.260	B
P2	4	44.44	25000	445000	91.395	B
P4	5	55.56	18000	463000	95.091	C
P3	6	66.67	12000	475000	97.556	C
P5	7	77.78	5500	480500	98.686	C
P7	8	88.89	4000	484500	99.507	C
P8	9	100.00	2400	486900	100.000	C

**Tabela F.3** – Exemplo de Análise ABC: Classificação ABC dos Produtos

Finalmente, estando na posse de todos os valores da percentagem do total de produtos assim como da percentagem do total do valor de uso acumulado pode então efectuar-se a classificação ou categorização ABC dos mesmos, tal como mostra a tabela E.3. Mais promenorizadamente e observando a tabela E.3, conclui-se que aos produtos P9 e P1 é atribuída a classe A pois correspondem a 11% e 20%, respectivamente, do total de artigos e que correspondem respectivamente a 51% e 78% do total de acumulado de valor de uso. No intervalo compreendido entre os 80 e 95% do acumulado total de valor de uso tem-se os produtos P6 e P2, da classe B, que correspondem, respectivamente, a 33% e 44% do total dos produtos analisados. Na classe C observam-se os produtos menos relevantes para o negócio, ou seja, os que se encontram no intervalo compreendido entre 95 e 100% do acumulado total de valor de uso e que pertecem a entre 40 e 100% do universo de produtos em causa, ou seja, os produtos P4, P3, P5, P7 e P8.



# ANEXO G

## Regras da Árvore de Decisão Gerada

---

As regras obtidas da árvore de decisão gerada pelo *software* CART após utilização do conjunto de treino obtido são então as seguintes:

```
/*Terminal Node 1*/
if( PROCURA <= 4.5 && COST_ACQ <= 9.21338 )
{
    terminalNode = -1;
    class = 3;
}

/*Terminal Node 2*/
if( COST_ACQ > 9.21338 && COST_ACQ <= 11.4728 && PROCURA <= 3.5 )
{
    terminalNode = -2;
    class = 3;
}

/*Terminal Node 3*/
if( COST_ACQ > 9.21338 && COST_ACQ <= 11.4728 && PROCURA > 3.5 && PROCURA <= 4.5 )
{
    terminalNode = -3;
    class = 2;
}

/*Terminal Node 4*/
if( PROCURA > 4.5 && PROCURA <= 7.5 && COST_ACQ <= 4.97941 )
{
    terminalNode = -4;
    class = 3;
}

/*Terminal Node 5*/
if( COST_ACQ > 4.97941 && COST_ACQ <= 5.73325 && PROCURA > 4.5 && PROCURA <= 6.5 )
{
    terminalNode = -5;
    class = 3;
}

/*Terminal Node 6*/
if( COST_ACQ > 4.97941 && COST_ACQ <= 5.73325 && PROCURA > 6.5 && PROCURA <= 7.5 )
{
    terminalNode = -6;
    class = 2;
}

/*Terminal Node 7*/
if( PROCURA > 4.5 && PROCURA <= 5.5 && COST_ACQ > 5.73325 && COST_ACQ <= 6.90125 )
{
    terminalNode = -7;
    class = 3;
}

/*Terminal Node 8*/
if( PROCURA > 4.5 && PROCURA <= 5.5 && COST_ACQ > 6.90125 && COST_ACQ <= 11.4728 )
{
    terminalNode = -8;
    class = 2;
}
```

```

/*Terminal Node 9*/
if( COST_ACQ > 5.73325 && COST_ACQ <= 11.4728 && PROCURA > 5.5 && PROCURA <= 7.5 )
{
    terminalNode = -9;
    class = 2;
}

/*Terminal Node 10*/
if( PROCURA <= 1.5 && COST_ACQ > 11.4728 && COST_ACQ <= 34.3247 )
{
    terminalNode = -10;
    class = 3;
}

/*Terminal Node 11*/
if( PROCURA <= 1.5 && COST_ACQ > 34.3247 )
{
    terminalNode = -11;
    class = 2;
}

/*Terminal Node 12*/
if( PROCURA > 1.5 && PROCURA <= 2.5 && COST_ACQ > 11.4728 && COST_ACQ <= 16.5267 )
{
    terminalNode = -12;
    class = 3;
}

/*Terminal Node 13*/
if( PROCURA > 1.5 && PROCURA <= 2.5 && COST_ACQ > 16.5267 && COST_ACQ <= 24.5527 )
{
    terminalNode = -13;
    class = 2;
}

/*Terminal Node 14*/
if( COST_ACQ > 11.4728 && COST_ACQ <= 24.5527 && PROCURA > 2.5 && PROCURA <= 7.5 )
{
    terminalNode = -14;
    class = 2;
}

/*Terminal Node 15*/
if( COST_ACQ > 24.5527 && COST_ACQ <= 38.893 && PROCURA > 1.5 && PROCURA <= 5.5 )
{
    terminalNode = -15;
    class = 2;
}

/*Terminal Node 16*/
if( COST_ACQ > 24.5527 && COST_ACQ <= 38.893 && PROCURA > 5.5 && PROCURA <= 7.5 )
{
    terminalNode = -16;
    class = 1;
}

/*Terminal Node 17*/
if( COST_ACQ > 38.893 && PROCURA > 1.5 && PROCURA <= 2.5 )
{
    terminalNode = -17;
    class = 2;
}

/*Terminal Node 18*/
if( COST_ACQ > 38.893 && PROCURA > 2.5 && PROCURA <= 7.5 )
{
    terminalNode = -18;
    class = 1;
}

/*Terminal Node 19*/
if( COST_ACQ <= 1.5865 && PROCURA > 7.5 && PROCURA <= 30 )
{
    terminalNode = -19;
    class = 3;
}

```

```

/*Terminal Node 20*/
if( COST_ACQ <= 1.5865 && PROCURA > 30 && PROCURA <= 41.5 )
{
    terminalNode = -20;
    class = 2;
}

/*Terminal Node 21*/
if( PROCURA > 7.5 && PROCURA <= 12.5 && COST_ACQ > 1.5865 && COST_ACQ <= 3.10643 )
{
    terminalNode = -21;
    class = 3;
}

/*Terminal Node 22*/
if( COST_ACQ > 3.10643 && COST_ACQ <= 3.72875 && PROCURA > 7.5 && PROCURA <= 9.5 )
{
    terminalNode = -22;
    class = 3;
}

/*Terminal Node 23*/
if( COST_ACQ > 3.10643 && COST_ACQ <= 3.72875 && PROCURA > 9.5 && PROCURA <= 12.5 )
{
    terminalNode = -23;
    class = 2;
}

/*Terminal Node 24*/
if( PROCURA > 7.5 && PROCURA <= 12.5 && COST_ACQ > 3.72875 && COST_ACQ <= 5.73325 )
{
    terminalNode = -24;
    class = 2;
}

/*Terminal Node 25*/
if( COST_ACQ > 1.5865 && COST_ACQ <= 1.95517 && PROCURA > 12.5 && PROCURA <= 17.5 )
{
    terminalNode = -25;
    class = 3;
}

/*Terminal Node 26*/
if( COST_ACQ > 1.5865 && COST_ACQ <= 1.95517 && PROCURA > 17.5 && PROCURA <= 35.5 )
{
    terminalNode = -26;
    class = 2;
}

/*Terminal Node 27*/
if( PROCURA > 12.5 && PROCURA <= 35.5 && COST_ACQ > 1.95517 && COST_ACQ <= 5.73325 )
{
    terminalNode = -27;
    class = 2;
}

/*Terminal Node 28*/
if( PROCURA > 35.5 && PROCURA <= 41.5 && COST_ACQ > 1.5865 && COST_ACQ <= 4.28232 )
{
    terminalNode = -28;
    class = 2;
}

/*Terminal Node 29*/
if( PROCURA > 35.5 && PROCURA <= 41.5 && COST_ACQ > 4.28232 && COST_ACQ <= 5.73325 )
{
    terminalNode = -29;
    class = 1;
}

/*Terminal Node 30*/
if( COST_ACQ > 5.73325 && COST_ACQ <= 11.4475 && PROCURA > 7.5 && PROCURA <= 15.5 )
{
    terminalNode = -30;
    class = 2;
}

```

```
/*Terminal Node 31*/
if( PROCURA > 15.5 && PROCURA <= 21 && COST_ACQ > 5.73325 && COST_ACQ <= 8.9144
{
    terminalNode = -31;
    class = 2;
}

/*Terminal Node 32*/
if( PROCURA > 15.5 && PROCURA <= 21 && COST_ACQ > 8.9144 && COST_ACQ <= 11.4475 )
{
    terminalNode = -32;
    class = 1;
}

/*Terminal Node 33*/
if( COST_ACQ > 5.73325 && COST_ACQ <= 11.4475 && PROCURA > 21 && PROCURA <= 41.5 )
{
    terminalNode = -33;
    class = 1;
}

/*Terminal Node 34*/
if( COST_ACQ <= 1.49625 && PROCURA > 41.5 && PROCURA <= 140 )
{
    terminalNode = -34;
    class = 2;
}

/*Terminal Node 35*/
if( COST_ACQ <= 1.49625 && PROCURA > 140 )
{
    terminalNode = -35;
    class = 1;
}

/*Terminal Node 36*/
if( COST_ACQ > 1.49625 && COST_ACQ <= 2.6125 && PROCURA > 41.5 && PROCURA <= 71.5 )
{
    terminalNode = -36;
    class = 2;
}

/*Terminal Node 37*/
if( COST_ACQ > 1.49625 && COST_ACQ <= 2.6125 && PROCURA > 71.5 )
{
    terminalNode = -37;
    class = 1;
}

/*Terminal Node 38*/
if( PROCURA > 41.5 && COST_ACQ > 2.6125 && COST_ACQ <= 11.4475 )
{
    terminalNode = -38;
    class = 1;
}

/*Terminal Node 39*/
if( PROCURA > 7.5 && PROCURA <= 10.5 && COST_ACQ > 11.4475 && COST_ACQ <= 16.7913 )
{
    terminalNode = -39;
    class = 2;
}

/*Terminal Node 40*/
if( PROCURA > 7.5 && PROCURA <= 10.5 && COST_ACQ > 16.7913 )
{
    terminalNode = -40;
    class = 1;
}

/*Terminal Node 41*/
if( COST_ACQ > 11.4475 && PROCURA > 10.5 )
{
    terminalNode = -41;
    class = 1;
}
```